FileMaker® Server 12

Custom Web Publishing mit XML

© 2007-2012 FileMaker, Inc. Alle Rechte vorbehalten.

FileMaker, Inc.
5201 Patrick Henry Drive
Santa Clara, California 95054, USA

FileMaker und Bento sind Marken von FileMaker, Inc., eingetragen in den USA und anderen Ländern. Das Dateiordner-Logo und das Bento-Logo sind Marken von FileMaker, Inc. Alle anderen Marken sind Eigentum der jeweiligen Besitzer.

Die FileMaker-Dokumentation ist urheberrechtlich geschützt. Sie dürfen diese Dokumentation ohne schriftliche Genehmigung von FileMaker weder vervielfältigen noch verteilen. Diese Dokumentation darf ausschließlich mit einer gültigen, lizenzierten Kopie der FileMaker-Software verwendet werden.

Alle in den Beispielen erwähnten Personen, Firmen, E-Mail-Adressen und URLs sind rein fiktiv und jegliche Ähnlichkeit mit bestehenden Personen, Firmen, E-Mail-Adressen und URLs ist rein zufällig. Die Danksagungen und Urheberrechtshinweise finden Sie im entsprechenden Dokument, das mit der Software geliefert wurde. Die Erwähnung von Produkten und URLs Dritter dient nur zur Information und stellt keine Empfehlung dar. FileMaker, Inc. übernimmt keine Verantwortung für die Leistung dieser Produkte.

Weitere Informationen finden Sie auf unserer Website unter www.filemaker.de.

Edition: 01

Inhalt

Vorwort 6
Über dieses Handbuch 6

Kapitel 1
Einführung in Custom Web Publishing 7
Über die Web Publishing Engine 8
Verarbeitung einer Web Publishing Engine -Anforderung 8
Custom Web Publishing mit PHP 9
Custom Web Publishing mit XML 9
Vergleich von PHP mit XML 9
Gründe für PHP 9
Gründe für XML 9

Kapitel 2
Erläuterung von Custom Web Publishing mit XML 11
Erstellen dynamischer Webseiten mit der Web Publishing Engine 11
Wichtige Funktionen in Custom Web Publishing mit XML 11
Voraussetzungen für Web Publishing 12
Voraussetzungen für das Veröffentlichen einer Datenbank mit Custom Web Publishing 12
Voraussetzungen für Web-Benutzer zum Zugriff auf eine Custom Web Publishing-Lösung 12
Verbindung zum Internet oder zu einem Intranet 13
Die nächsten Schritte 13

Kapitel 3
Vorbereiten von Datenbanken für Custom Web Publishing 14
Aktivieren von Instant Web Publishing in einer Datenbank 14
Zugriff auf eine geschützte Datenbank 14
Schützen veröffentlichter Datenbanken 15
Web Server-Unterstützung für Internet-Medientypen (MIME) 16
Veröffentlichen des Inhalts von Containerfeldern im Web 16
In eine Datenbank eingebettete Containerfeld-Objekte 16
Containerfelder mit referenzierten Dateien 16
Containerfelder mit extern gespeicherten Daten 17
Anzeige von Containerfelddaten durch Web-Benutzer 18
FileMaker-Scripts und Custom Web Publishing 18
Tipps und Überlegungen zu Scripts 19
Scriptverhalten in Custom Web Publishing-Lösungen 20
Script-Trigger in Custom Web Publishing-Lösungen 21

Kapitel 4
Zugriff auf XML-Daten mit der Web Publishing Engine 22
Verwenden von Custom Web Publishing mit XML 22
Unterschiede zwischen der Web Publishing Engine und FileMaker Pro XML-Import/Export 22
Wie die Web Publishing Engine aus einer Abfrage XML-Daten generiert 23
Allgemeines Verfahren beim Zugriff auf XML-Daten von der Web Publishing Engine 24

Erläuterung der URL-Syntax für XML-Daten und Containerobjekte 24
Erläuterung der URL-Syntax für XML-Daten 25
Erläuterung der URL-Syntax für FileMaker-Containerobjekte in XML-Lösungen 25
Erläuterung von URL-Textkodierung 26
Zugriff auf XML-Daten über die Web Publishing Engine 27
Erläuterung von Namensräumen für FileMaker XML 27
Erläuterung der FileMaker-Datenbankfehlercodes 28
Abrufen der Document Type Definitions für die FileMaker-Grammatiken 28
Verwenden der fmresultset-Grammatik 29
Beschreibung der Elemente der fmresultset-Grammatik 29
Beispiel für XML-Daten in der fmresultset-Grammatik 31
Verwenden anderer FileMaker XML-Grammatiken 32
Beschreibung der Elemente in der FMPXMLRESULT-Grammatik 32
Beispiel für XML-Daten in der FMPXMLRESULT-Grammatik 33
Beschreibung der Elemente der Grammatik FMPXMLLAYOUT 34
XML-Daten in der FMPXMLLAYOUT-Grammatik – Beispiel 36
Erläuterung von UTF-8-kodierten Daten 37
Abfragen von XML-Daten mithilfe von FileMaker-Query-Strings 37
Wechseln des Layouts für eine XML-Antwort 39
Erläuterung der Verarbeitung einer XML-Abfrage 39
Fehlerbehebung beim XML-Dokumentzugriff 40

Kapitel 5
Einstellen, Testen und Überwachen einer Site 41
Einstellen einer Custom Web Publishing-Site 41
Testen einer Custom Web Publishing-Site 42
Beispiele für Stylesheets zum Testen der XML-Ausgabe 42
Überwachen Ihrer Website 43
Verwenden der Web-Server-Zugriffs- und Fehlerprotokolle 43
Verwenden des Web Publishing Engine-Protokolls 44
Verwenden des Web-Server-Modul-Fehlerprotokolls 45
Verwenden der Tomcat-Protokolle 46

Anhang A
Gültige Namen in Query-Strings 47
Erläuterung von Query-Befehlen und -Parametern 47
Richtlinien für die Verwendung von Query-Befehlen und -Parametern 47
Query-Befehls-Parsing 48
Erläuterung der Syntax für einen voll qualifizierten Feldnamen 49
Verwenden von Query-Befehlen für Ausschnittfelder 49
Erläuterung der Syntax für das Angeben eines Variablenfelds 51
Query-Befehlsreferenz 52
Query-Befehl –dbnames (Datenbanknamen) 52
Query-Befehl –delete (Datensatz löschen) 52
Query-Befehl –dup (Datensatz duplizieren) 52
Query-Befehl –edit (Datensatz bearbeiten) 52
Query-Befehle –find, –findall oder –findany (Datensätze suchen) 53
–findquery (Compound-Suche, Query-Befehl) 53

–layoutnames (Layoutnamen) (Query-Befehl) 54
–new (Neuer Datensatz) (Query-Befehl) 54
–scriptnames (Scriptnamen) (Query-Befehl) 54
Query-Befehl –view (Layoutinformationen anzeigen) 55
Query-Parameterreferenz 55
Query-Parameter –db (Datenbankname) 55
–delete.related (Löschen von Ausschnittdatensätzen, Query-Parameter) 55
Query-Parameter –field (Containerfeldname) 56
Query-Parameter „fieldname“ (Nicht-Containerfeldname) 56
Query-Parameter „fieldname.Op“ (Vergleichsoperator) 57
Query-Parameter –lay (Layout) 58
Query-Parameter –lay.response (Layout wechseln für Antwort) 58
Query-Parameter –lop (Logischer Operator) 58
Query-Parameter –max (Maximum Datensätze) 58
Query-Parameter –modid (Änderungs-ID) 59
–query (Compound-Suchabfrage, Query-Parameter) 59
Query-Parameter –recid (Datensatz-ID) 60
–relatedsets.filter (Ausschnittdatensätze filtern, Query-Parameter) 61
–relatedsets.max (Ausschnittdatensätze beschränken, Query-Parameter) 62
–script (Script) (Query-Parameter) 62
Query-Parameter –script.param (Parameter an Script übergeben) 62
Query-Parameter –script.prefind (Script vor Suchen) 63
Query-Parameter –script.prefind.param (Parameter an Script vor Suchen übergeben) 63
Query-Parameter –script.presort (Script vor Sortieren) 63
Query-Parameter –script.presort.param (Parameter an Script vor Sortieren übergeben) 64
Query-Parameter –skip (Datensätze überspringen) 64
Query-Parameter –sortfield (Sortierfeld) 64
Query-Parameter –sortorder (Sortierfolge) 65

Anhang B
Fehlercodes für Custom Web Publishing 67
Fehlercodenummern in XML-Format 67
Fehlercodenummern für FileMaker-Datenbanken 67

Anhang C
Änderungen bei XML-Queries in FileMaker 12 75
XML-Query-Änderungen in der Syntax 75
XML-Query-Änderungen in der Semantik 75
Unterschiede im Query-Parsing 75
Unterschiede in der Query-Verarbeitung 76
Unterschiede in den zurückgegebenen Fehlercodes 76

Index 78

Vorwort

Über dieses Handbuch

Dieses Handbuch setzt voraus, dass Sie mit XML, der Entwicklung von Websites und dem Einsatz von FileMaker® Pro zur Erstellung von Datenbanken vertraut sind. Sie sollten mit den Grundlagen der Datenbankgestaltung von FileMaker Pro vertraut sein und die Konzepte von Feldern, Beziehungen, Layouts, Ausschnitten und Containerfeldern verstehen. Dieses Handbuch enthält die folgenden Informationen über Custom Web Publishing mit XML auf FileMaker Server:

- Voraussetzungen für die Entwicklung einer Custom Web Publishing-Lösung mit XML
- Veröffentlichen Ihrer Datenbanken mit XML
- Voraussetzungen für Web-Benutzer zum Zugriff auf eine Custom Web Publishing-Lösung
- Abruf von XML-Daten aus Datenbanken, die FileMaker Server bereitstellt

Wichtig Sie können PDF-Dateien der FileMaker-Dokumentation unter http://www.filemaker.de/support/product/documentation.html herunterladen. Aktualisierungen dieses Dokuments erhalten Sie ebenfalls auf der Website.

Die Dokumentation für FileMaker Server umfasst die folgenden Informationen:

Weitere Informationen über	**Siehe**
Installieren und Konfigurieren von FileMaker Server	*FileMaker Server Einführung* FileMaker Server Hilfe
Instant Web Publishing	*FileMaker Instant Publishing Handbuch*
Custom Web Publishing mit PHP	*FileMaker Server Custom Web Publishing mit PHP*
Custom Web Publishing mit XML	*FileMaker Server Custom Web Publishing mit XML* (dieses Handbuch)
Installation und Konfiguration der ODBC- und JDBC-Treiber und Verwendung von ODBC und JDBC	*FileMaker ODBC- und JDBC-Handbuch*

Kapitel 1
Einführung in Custom Web Publishing

Mit FileMaker Server können Sie Ihre FileMaker-Datenbank im Internet oder in einem Intranet wie folgt veröffentlichen:

Instant Web Publishing: Mit Instant Web Publishing können Sie Ihre Datenbank schnell und einfach im Web veröffentlichen. Sie müssen Ihre Datenbankdateien dazu nicht verändern oder zusätzliche Software installieren – jeder Benutzer mit einem kompatiblen Webbrowser und Zugriff auf das Internet bzw. ein Intranet kann eine Verbindung zu Ihrer Datenbank herstellen, um Datensätze anzuzeigen, zu bearbeiten, zu sortieren oder zu durchsuchen, wenn Sie ihm die entsprechenden Zugriffsrechte geben.

Für Instant Web Publishing muss der Host-Computer FileMaker Pro, FileMaker Pro Advanced oder FileMaker Server Advanced ausführen. Die Benutzeroberfläche ähnelt der FileMaker Pro-Desktop-Anwendung. Die Web-Seiten und Formulare, mit denen der Web-Benutzer arbeitet, hängen von den in der FileMaker Pro-Datenbank definierten Layouts und Ansichten ab. Weitere Informationen finden Sie im *FileMaker Instant Web Publishing Handbuch***.**

Statisches Veröffentlichen: Wenn sich Ihre Daten selten ändern oder Sie keine Live-Verbindung Ihrer Benutzer zu Ihrer Datenbank wünschen, können Sie die Daten statisch veröffentlichen. Bei der statischen Veröffentlichung exportieren Sie Daten aus einer FileMaker Pro-Datenbank, um eine Web-Seite zu erstellen, die Sie anhand von HTML weiter anpassen können. Die Web-Seite ändert sich nicht, wenn sich Informationen in Ihrer Datenbank ändern, und die Benutzer stellen keine Verbindung zu Ihrer Datenbank her. (Mit Instant Web Publishing werden die Daten in einem Webbrowser-Fenster jedes Mal aktiviert, wenn der Browser eine Anfrage an FileMaker Server sendet.) Weitere Informationen finden Sie im *FileMaker Instant Web Publishing Handbuch*.

Custom Web Publishing: Erscheinungsbild und Funktionalität Ihrer veröffentlichten Datenbank können Sie über die für FileMaker Server verfügbaren Technologien für Custom Web Publishing beeinflussen. Für FileMaker Server, der die veröffentlichten Datenbanken bereitstellt, muss weder FileMaker Pro installiert noch Custom Web Publishing verfügbar sein.

Mit Custom Web Publishing können Sie:

- Ihre Datenbank in andere Web-Sites integrieren.
- festlegen, wie Benutzer mit den Daten arbeiten.
- steuern, wie Daten in Webbrowsern angezeigt werden.

FileMaker Server stellt zwei Custom Web Publishing-Technologien zur Verfügung:

- Custom Web Publishing mit PHP: Verwenden Sie das FileMaker API for PHP, das eine objektorientierte PHP-Schnittstelle zu FileMaker Pro-Datenbanken bereitstellt, um Ihre FileMaker-Daten in einer PHP-Web-Anwendung zu integrieren. Da Sie die PHP-Webseiten selbst programmieren, haben Sie die komplette Kontrolle über die Benutzeroberfläche und wie der Benutzer mit den Daten arbeitet.
- Custom Web Publishing mit XML: Verwenden Sie XML Data Publishing, um FileMaker-Daten mit anderen Webseiten und Anwendungen auszutauschen. Indem Sie HTTP-URL-Abfragen mit FileMaker-Query-Befehlen und -Parametern verwenden, können Sie eine Datenbank abfragen, die von FileMaker Server bereitgestellt wird, die resultierenden Daten in XML-Format herunterladen und die resultierenden XML-Daten so verwenden, wie Sie es möchten.

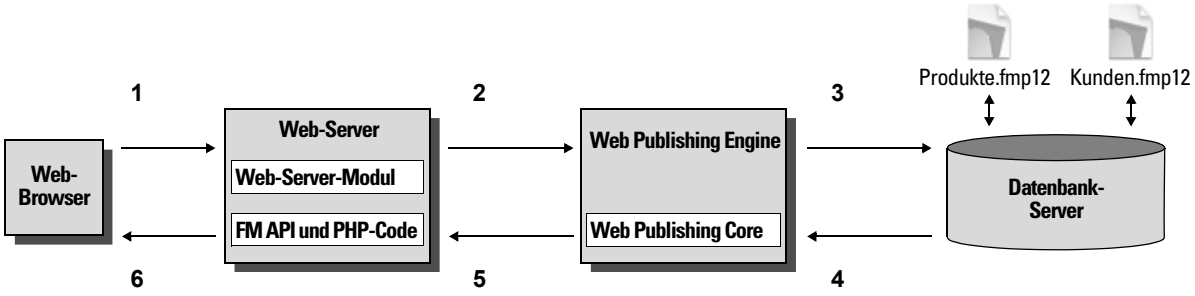
Über die Web Publishing Engine

Um Instant Web Publishing und Custom Web Publishing zu unterstützen, verwendet FileMaker Server einen Satz von Softwarekomponenten, die sogenannte *FileMaker Server Web Publishing Engine*. Die Web Publishing Engine verwaltet Interaktionen zwischen dem Browser eines Web-Benutzers, Ihrem Web-Server und FileMaker Server.

Custom Web Publishing mit XML: Web-Benutzer greifen auf Ihre Custom Web Publishing-Lösung zu, indem sie auf einen HREF-Link klicken oder einen Uniform Resource Locator (URL) eingeben, der die Web-Server-Adresse und eine FileMaker-Query-Zeichenfolgenabfrage angibt. Die Web Publishing Engine gibt die in der Query-Zeichenfolgenabfrage angegebenen XML-Daten zurück.

Custom Web Publishing mit PHP: Wenn ein Web-Benutzer auf Ihre Custom Web Publishing-Lösung zugreift, stellt PHP auf FileMaker Server eine Verbindung mit der Web Publishing Engine her und reagiert über das FileMaker API for PHP.

Verwenden der FileMaker Server Web Publishing Engine für Custom Web Publishing



Verarbeitung einer Web Publishing Engine -Anforderung

1. Eine Anforderung wird von einem Webbrowser oder Programm an den Web-Server gesendet.
2. Der Web-Server leitet die Anforderung über das FileMaker Web-Server-Modul an die Web Publishing Engine weiter.
3. Die Web Publishing Engine fordert Daten von der Datenbank an, die der Datenbank-Server bereitstellt.
4. FileMaker Server sendet die angeforderten FileMaker-Daten an die Web Publishing Engine.
5. Die Web Publishing Engine konvertiert die FileMaker-Daten, um die Anforderung zu beantworten.
   - Für PHP-Anforderungen antwortet die Web Publishing Engine auf die API-Anforderung.
   - Für XML-Anforderungen sendet die Web Publishing Engine die XML-Daten direkt an den Web-Server.
6. Der Web-Server sendet die Ausgabe an den anfordernden Webbrowser oder das anfordernde Programm.

Wichtig Sicherheitsüberlegungen sind wichtig, wenn Sie Daten im Web veröffentlichen. Sehen Sie sich die Sicherheitsrichtlinien im *FileMaker Pro Benutzerhandbuch* an, das als PDF-Datei unter http://www.filemaker.de/support/product/documentation.html verfügbar ist.

Custom Web Publishing mit PHP

Das FileMaker API for PHP bietet eine objektorientierte PHP-Schnittstelle zu FileMaker-Datenbanken. Mithilfe des FileMaker API for PHP kann sowohl auf Daten als auch auf Logik, die in einer FileMaker Pro-Datenbank gespeichert sind, zugegriffen werden. Diese können auch im Internet veröffentlicht oder an andere Anwendungen exportiert werden. Das API unterstützt zudem komplexe und zusammengesetzte Suchabfragen für das Extrahieren und Filtern von in FileMaker Pro-Datenbanken gespeicherten Daten.

Ursprünglich als prozedurale Programmiersprache entwickelt, wurde PHP als objektorientierte Programmiersprache für die Web-Entwicklung ausgebaut. PHP bietet Programmiersprachenfunktionalität für den Aufbau praktisch jeder Art von Logik innerhalb einer Site-Seite. Zum Beispiel können Sie bedingte Logikkonstrukte verwenden, um die Seitengenerierung, das Datenrouting oder den Workflow zu steuern. PHP bietet zudem Funktionen für Site-Administration und Sicherheit.

Custom Web Publishing mit XML

FileMaker Custom Web Publishing mit XML ermöglicht Ihnen, Abfrageanforderungen an eine FileMaker Pro-Datenbank zu senden, die von FileMaker Server bereitgestellt wird, und die resultierenden Daten anzuzeigen und zu ändern. Mithilfe einer HTTP-Abfrage mit den geeigneten Query-Befehlen und -Parametern können Sie FileMaker-Daten als ein XML-Dokument abrufen. Sie können die XML-Daten in andere Anwendungen exportieren.

Vergleich von PHP mit XML

Der folgende Abschnitt zeigt einige Richtlinien auf, um die beste Lösung für Ihre Site zu bestimmen.

Gründe für PHP

- PHP ist eine leistungsstarke, objektorientierte prozedurale Scriptingsprache und relativ leicht zu erlernen. Es gibt viele Ressourcen für Schulung, Entwicklung und Support.
- Mithilfe des FileMaker API for PHP kann sowohl auf Daten als auch auf Logik, die in einer FileMaker Pro-Datenbank gespeichert sind, zugegriffen werden. Diese können auch im Internet veröffentlicht oder an andere Anwendungen exportiert werden.
- Mit PHP können Sie bedingte Logik verwenden, um den Seitenaufbau oder den Fluss zu kontrollieren.
- PHP bietet Programmiersprachenfunktionalität für den Aufbau vieler Arten von Logik innerhalb einer Site-Seite.
- PHP ist eine der beliebtesten Web-Scriptingsprachen.
- PHP ist eine Open-Source-Sprache, verfügbar unter http://php.net.
- PHP ermöglicht den Zugriff auf eine große Vielzahl von Third-Party-Komponenten, die Sie in Ihre Lösungen integrieren können.

Hinweis Weitere Informationen über Custom Web Publishing mit PHP finden Sie im Handbuch *FileMaker Server Custom Web Publishing mit PHP*.

Gründe für XML

- Die Syntax der FileMaker XML-Abfrageparameter ist für die Datenbankinteraktion konzipiert und vereinfacht die Lösungsentwicklung.
- XML ist ein W3C-Standard.

- XML ist ein von Maschinen und Menschen lesbares Format, das Unicode unterstützt, so dass Daten in jeder geschriebenen Sprache kommuniziert werden können.
- XML ist bestens geeignet für die Darstellung von Datensätzen, Listen und Daten mit Baumstruktur.
- Sie können FMPXMLRESULT für den Zugriff auf XML-Daten mit Custom Web Publishing und für den XML-Export aus FileMaker Pro-Datenbanken verwenden.

Kapitel 2
Erläuterung von Custom Web Publishing mit XML

Erstellen dynamischer Webseiten mit der Web Publishing Engine

Die Web Publishing Engine bietet Custom Web Publishing für FileMaker Server mithilfe von XML-Datenveröffentlichung. Custom Web Publishing bietet mehrere Vorteile:

- **Anpassung**: Sie können festlegen, wie Web-Benutzer mit FileMaker-Daten interagieren und wie die Daten in Webbrowsern angezeigt werden.
- **Datenaustausch**: Mithilfe von FileMaker XML können Sie FileMaker-Daten mit anderen Webseiten und Programmen austauschen.
- **Datenintegration**: Sie können FileMaker-Daten in andere Websites, mit anderer Middleware und mit eigenen Anwendungen integrieren. Sie können die Daten so darstellen, als würden sie zu einer anderen Website gehören, anstatt ein vollständiges FileMaker-Layout im Webbrowser anzuzeigen.
- **Sicherheit**: Der FileMaker Server-Administrator kann Instant Web Publishing oder XML Web Publishing einzeln für alle Datenbanken, die der Server bereitstellt, aktivieren oder deaktivieren. Als Eigentümer der FileMaker-Datenbank können Sie den Benutzerzugriff auf Instant Web Publishing oder XML Web Publishing für jede Datenbank steuern.
- **Steuern und Filtern von veröffentlichten Daten**: Sie können die Daten und die Art der Datenbankinformationen, die Sie veröffentlichen wollen, steuern und filtern und damit unbefugte Nutzung der Datenbank verhindern. Sie können auch Metadaten ausblenden wie z. B. Datenbank- und Feldnamen.
- **Basierend auf offenen Standards**: Ihnen stehen mehr Tools, Ressourcen und Fachpersonal für Custom Web Publishing-Lösungen zur Verfügung. Wenn Sie Standard-XML kennen, können Sie sofort mit der Entwicklung von Lösungen beginnen, nachdem Sie ein paar spezielle Details über Custom Web Publishing mit XML gelernt haben, z. B. die erforderliche URL-Syntax und Query-Parameter.

Mit Custom Web Publishing mit XML können Sie Daten aus FileMaker-Datenbanken abrufen und die Daten einfach in anderen Ausgabeformaten verwenden. Mithilfe einer HTTP-Abfrage mit den geeigneten Query-Befehlen und -Parametern können Sie FileMaker-Daten als ein XML-Dokument abrufen. Sie können die XML-Daten in anderen Anwendungen verwenden. Weitere Informationen finden Sie unter „Zugriff auf XML-Daten über die Web Publishing Engine“ auf Seite 27.

Wichtige Funktionen in Custom Web Publishing mit XML

FileMaker Server Custom Web Publishing mit XML bietet mehrere wichtige neue Funktionen:

- Datenbanken werden auf FileMaker Server bereitgestellt und FileMaker Pro muss nicht ausgeführt werden.
- Sie können die Server-seitige Verarbeitung von XML mit JavaScript nutzen.

- Wie in FileMaker Pro basiert der Zugriff auf Daten, Layouts und Felder auf den Benutzerkontoeinstellungen, die in den Zugriffsrechten der Datenbank definiert sind. Die Web Publishing Engine unterstützt auch mehrere andere Sicherheitsverbesserungen. Weitere Informationen finden Sie unter „Schützen veröffentlichter Datenbanken“ auf Seite 15.
- Web-Benutzer können komplexe Scripts mit mehreren Schritten ausführen. FileMaker unterstützt ungefähr 65 Scriptschritte in Custom Web Publishing. Informationen finden Sie im Abschnitt „FileMaker-Scripts und Custom Web Publishing“ auf Seite 18.
- Sie können einen Parameterwert an ein FileMaker-Script übergeben. Weitere Informationen finden Sie unter „Query-Parameter –script.param (Parameter an Script übergeben)“ auf Seite 62, „Query-Parameter –script.prefind.param (Parameter an Script vor Suchen übergeben)“ auf Seite 63 und „Query-Parameter –script.presort.param (Parameter an Script vor Sortieren übergeben)“ auf Seite 64.
- Mit der `fmresultset`-XML-Grammatik können Sie nach Name auf Felder zugreifen und `relatedset`- (Ausschnitt) Daten manipulieren.
- Für den Zugriff auf Daten in einer Datenbank müssen Sie ein Layout angeben. Weitere Informationen finden Sie unter Anhang A, „Gültige Namen in Query-Strings“.

Voraussetzungen für Web Publishing

Voraussetzungen für das Veröffentlichen einer Datenbank mit Custom Web Publishing

Um Datenbanken mithilfe von Custom Web Publishing mit XML zu veröffentlichen, benötigen Sie:

- einen FileMaker Server-Einsatz, der Folgendes umfasst:
  - einen Web-Server, entweder Microsoft IIS (Windows) oder Apache (Mac OS X)
  - den FileMaker Datenbank-Server, aktiviert für Custom Web Publishing
  - die Web Publishing Engine, installiert und konfiguriert
- eine oder mehrere FileMaker Pro-Datenbanken, die FileMaker Server bereitstellt
- IP-Adresse oder Domänenname des Hosts, auf dem der Web-Server läuft
- einen beliebigen Webbrowser und Zugriff auf den Web-Server, um Ihre Custom Web Publishing-Lösung zu entwickeln und zu testen

Weitere Informationen finden Sie im *Handbuch FileMaker Server Einführung*.

Voraussetzungen für Web-Benutzer zum Zugriff auf eine Custom Web Publishing-Lösung

Für den Zugriff auf eine Custom Web Publishing-Lösung mit XML benötigen Web-Benutzer:

- einen Webbrowser
- Zugang zum Internet oder zu einem Intranet und zum Web-Server
- IP-Adresse oder Domänenname des Hosts, auf dem der Web-Server läuft

Wenn die Datenbank passwortgeschützt ist, müssen Web-Benutzer auch einen Benutzernamen und ein Passwort für ein Datenbankkonto eingeben.

Verbindung zum Internet oder zu einem Intranet

Wenn Sie Datenbanken im Internet oder in einem Intranet veröffentlichen, muss auf dem Host-Computer FileMaker Server laufen und die freizugebenden Datenbanken müssen bereitgestellt und verfügbar sein. Zusätzlich:

- Veröffentlichen Sie Ihre Datenbank auf einem Computer, der über eine ständige Verbindung zum Internet bzw. Intranet verfügt. Sie können Datenbanken zwar ohne ständige Verbindung veröffentlichen, aber sie stehen Web-Benutzern nur zur Verfügung, wenn Ihr Computer mit dem Internet bzw. Intranet verbunden ist.
- Der Host-Computer für den Web-Server, der Teil des FileMaker Server-Einsatzes ist, muss über eine eigene *statische* (permanente) IP-Adresse oder einen Domänennamen verfügen. Wenn Sie die Verbindung zum Internet über einen Internet-Anbieter (ISP) herstellen, könnte Ihre IP-Adresse *dynamisch zugewiesen* werden (d. h., sie ändert sich bei jeder Verbindung). Mit einer dynamischen IP-Adresse ist es für Web-Benutzer schwieriger, Ihre Datenbanken zu finden. Wenn Sie sich über Ihre Zugangsart nicht sicher sind, wenden Sie sich an Ihren ISP oder Netzwerkadministrator.

Die nächsten Schritte

Hier einige Vorschläge, um mit der Entwicklung von Custom Web Publishing-Lösungen zu beginnen:

- Falls noch nicht geschehen, aktivieren Sie Custom Web Publishing mithilfe der FileMaker Server Admin Console. Informationen hierzu finden Sie in der FileMaker Server Hilfe und im *Handbuch FileMaker Server Einführung*.
- Öffnen Sie in FileMaker Pro jede FileMaker-Datenbank, die Sie veröffentlichen wollen, und stellen Sie sicher, dass in der Datenbank die geeigneten erweiterten Zugriffsrechte für Custom Web Publishing aktiviert sind Weitere Informationen finden Sie unter „Aktivieren von Instant Web Publishing in einer Datenbank“ auf Seite 14.
- Wie Sie mithilfe von XML auf Daten in FileMaker-Datenbanken zugreifen, erfahren Sie in „Zugriff auf XML-Daten über die Web Publishing Engine“ auf Seite 27.

Kapitel 3
Vorbereiten von Datenbanken für Custom Web Publishing

Bevor Sie Custom Web Publishing mit einer Datenbank verwenden können, müssen Sie die Datenbank vorbereiten und vor unbefugtem Zugriff schützen.

Aktivieren von Instant Web Publishing in einer Datenbank

Sie müssen Custom Web Publishing mit XML in jeder Datenbank aktivieren, die Sie veröffentlichen wollen. Wenn Sie Custom Web Publishing mit XML in der Datenbank nicht aktivieren, können Web-Benutzer Custom Web Publishing nicht nutzen, um auf die Datenbank zuzugreifen, selbst wenn sie von FileMaker Server bereitgestellt ist, der für eine Web Publishing Engine konfiguriert wurde.

So aktivieren Sie Custom Web Publishing in einer Datenbank:

1. Öffnen Sie in FileMaker Pro die zu veröffentlichende Datenbank mit einem Konto, das über die Berechtigung für vollen Zugriff verfügt. Alternativ können Sie die Datenbank mit einem Konto öffnen, das über die Berechtigung „Erweiterte Zugriffsrechte verwalten“ verfügt.
2. Weisen Sie das erweiterte Zugriffsrecht „Custom Web Publishing mit XML“ durch Verwendung dieses Schlüsselworts zu: **fmxml**
3. Weisen Sie die Berechtigung(en), die das erweiterte Custom Web Publishing mit XML-Zugriffsrecht enthält, einem oder mehreren Konten bzw. dem Admin- oder Gastkonto zu.

Hinweis Wenn Sie Kontonamen und Passwörter für Custom Web Publishing-Lösungen definieren, verwenden Sie druckbare ASCII-Zeichen. Beispiel: **a**-**z**, **A**-**Z** und **0**-**9**. Verwenden Sie für sicherere Kontonamen und Passwörter zudem Satzzeichen wie „**!**“ und „**%**“, aber verwenden Sie keine Doppelpunkte. Informationen über das Einrichten von Konten finden Sie in der FileMaker Pro Hilfe.

Zugriff auf eine geschützte Datenbank

Wenn Web-Benutzer eine Custom Web Publishing-Lösung für den Zugriff auf eine Datenbank verwenden, werden sie eventuell aufgefordert, ihre Kontoinformationen einzugeben. Wenn das Gastkonto für die Datenbank deaktiviert ist oder über keine Berechtigung mit dem erweiterten Zugriffsrecht für Custom Web Publishing verfügt, verwendet die Web Publishing Engine die HTTP-Standardauthentifizierung, um entsprechende Informationen von Web-Benutzern zu verlangen. Der Browser des Web-Benutzers zeigt das Dialogfeld für die HTTP-Standardauthentifizierung an, in dem der Benutzer einen Benutzernamen und ein Passwort für ein Konto eingibt, das über ein erweitertes Zugriffsrecht für Custom Web Publishing verfügt.

Die folgende Liste fasst die Abläufe zusammen, wenn ein Web-Benutzer eine Custom Web Publishing-Lösung für den Zugriff auf eine Datenbank verwendet:

- Wenn Sie einem Konto kein Passwort zugewiesen haben, geben Web-Benutzer nur den Kontonamen an.

- Wenn das Gastkonto deaktiviert ist, werden die Benutzer beim Zugriff auf die Datenbank aufgefordert, Kontoname und Passwort anzugeben. Für das Konto muss ein erweitertes Zugriffsrecht für Custom Web Publishing aktiviert sein.
- Wenn das Gastkonto aktiviert ist und über eine Berechtigung mit einem erweiterten Zugriffsrecht für Custom Web Publishing verfügt, öffnen automatisch alle Web-Benutzer die Datenbank mit den Zugriffsrechten, die dem Gastkonto zugewiesen sind. Wenn das erweiterte Zugriffsrecht für Custom Web Publishing dem Gastkonto zugewiesen ist:
  - Web-Benutzer werden nicht aufgefordert, Kontoname und Passwort einzugeben, wenn sie eine Datei öffnen.
  - Alle Web-Benutzer werden automatisch mit dem Gastkonto angemeldet und erhalten die Gast-Zugriffsrechte. Sie können Benutzern über den Scriptschritt „Erneut anmelden“ erlauben, ihre Anmeldekonten von einem Webbrowser aus zu wechseln (z. B. vom Gastkonto zu einem anderen Konto mit mehr Rechten).
  - Die Standardberechtigungen für Gastkonten umfassen den Nur-Lese-Zugriff. Sie können für dieses Konto die Standardberechtigungen einschließlich erweiterten Zugriffsrechten ändern. Weitere Informationen finden Sie in der FileMaker Pro Hilfe.

Hinweis Standardmäßig können Web-Benutzer ihr Kontopasswort nicht von einem Webbrowser aus ändern. Sie können diese Funktion über den Scriptschritt „Passwort ändern“ in Ihre Datenbank integrieren, damit Web-Benutzer ihr Passwort vom Browser aus ändern können. Weitere Informationen finden Sie in der FileMaker Pro Hilfe.

Schützen veröffentlichter Datenbanken

Wenn Sie Custom Web Publishing mit XML verwenden, können Sie beschränken, wer auf Ihre veröffentlichten Datenbanken zugreifen kann.

- Weisen Sie Datenbankkonten, die für Custom Web Publishing verwendet werden, Passwörter zu.
- Aktivieren Sie Custom Web Publishing mit XML nur in den Berechtigungen für Konten, denen Sie den Zugriff auf Ihre veröffentlichten Datenbanken erlauben wollen.
- Um einen Typ der Custom Web Publishing-Technologie für eine einzelne Datenbank zu aktivieren oder zu deaktivieren, wählen Sie das entsprechende erweiterte Zugriffsrecht.
- Aktivieren oder deaktivieren Sie einen Typ der Custom Web Publishing-Technologie für alle Custom Web Publishing-Lösungen in der Web Publishing Engine mit der FileMaker Server Admin Console. Weitere Informationen finden Sie in der FileMaker Server Hilfe.
- Konfigurieren Sie Ihren Web-Server, um die IP-Adressen einzuschränken, die über die Web Publishing Engine auf Ihre Datenbanken zugreifen können. Beispielsweise können Sie angeben, dass nur Web-Benutzer von der IP-Adresse 192.168.100.101 auf Ihre Datenbanken zugreifen können. Informationen über das Einschränken von IP-Adressen finden Sie in der Dokumentation zu Ihrem Web-Server.

- Verwenden Sie Secure Sockets Layer- (SSL) Verschlüsselung für die Kommunikation zwischen Ihrem Web-Server und den Browsern der Web-Benutzer. Die SSL-Verschlüsselung wandelt Informationen, die zwischen Servern und Clients übertragen werden, mithilfe von mathematischen Formeln in unverständliche Informationen um. Der englische Fachausdruck für diese Chiffrier-Algorithmen ist Ciphers. Diese Algorithmen nutzt der Empfänger, um mithilfe von Schlüsseln, den sogenannten Chiffrierschlüsseln, die Informationen wieder in verständliche Daten umzuwandeln. Informationen über das Aktivieren und Konfigurieren von SSL erhalten Sie in der Dokumentation zu Ihrem Web-Server.

Weitere Informationen zur Sicherung Ihrer Datenbank finden Sie im *FileMaker Pro Benutzerhandbuch*, das als PDF-Datei unter http://www.filemaker.de/support/product/documentation.html verfügbar ist.

Web Server-Unterstützung für Internet-Medientypen (MIME)

Ihr Web-Server bestimmt die Unterstützung für die aktuellen MIME-Typen (MIME; Multipurpose Internet Mail Extensions), die für das Internet registriert sind. Die Web Publishing Engine ändert die MIME-Unterstützung eines Web-Servers nicht. Weitere Informationen finden Sie in der Dokumentation zu Ihrem Web-Server.

Veröffentlichen des Inhalts von Containerfeldern im Web

Der Inhalt eines Containerfelds kann in die Datenbank eingebettet sein, per Verweis über einen relativen Pfad verknüpft sein oder extern gespeichert sein.

In eine Datenbank eingebettete Containerfeld-Objekte

Wenn ein Containerfeld die tatsächlichen Dateien in der FileMaker-Datenbank speichert, ist keine Aktion mit dem Inhalt des Containerfelds erforderlich, sofern die Datenbankdatei korrekt auf dem FileMaker Server freigegeben und zugänglich ist. Weitere Informationen finden Sie unter „Erläuterung der URL-Syntax für FileMaker-Containerobjekte in XML-Lösungen“ auf Seite 25.

Hinweis Die Web Publishing Engine unterstützt den progressiven Download von Audiodateien (.mp3), Videodateien (.mov, .mp4 und .avi empfohlen) und PDF-Dateien für interaktive Container. Zum Beispiel kann ein Web-Benutzer beginnen, einen Film anzusehen, auch wenn der Film noch nicht komplett heruntergeladen ist. Um den progressiven Download zuzulassen, müssen Sie die Dateien mithilfe von Optionen erstellen, die Streaming unterstützen oder für die Web-Anzeige optimiert wurden. Erstellen Sie beispielsweise Dateien mithilfe der Option „Für Web-Anzeige optimieren“.

Containerfelder mit referenzierten Dateien

Wenn ein Containerfeld einen Dateiverweis speichert, gehen Sie wie folgt vor, um die Dateien, auf die verwiesen wird, mit der Web Publishing Engine zu veröffentlichen:

So veröffentlichen Sie Containerfeldobjekte, die als Dateiverweise gespeichert sind:

1. Speichern Sie die Containerobjektdateien im Web-Ordner des FileMaker Pro-Ordners.
2. Fügen Sie in FileMaker Pro die Objekte in das Containerfeld ein und wählen Sie die Option **Nur Verweis auf die Datei speichern**.

3. Kopieren oder verschieben Sie die Objektdateien, auf die verwiesen wird, im Web-Ordner an die gleiche Stelle im relativen Pfad des Root-Ordners der Web-Server-Software.
   - **Für IIS (Windows):** `<Laufwerk>:\Inetpub\wwwroot`, wobei `<Laufwerk>` sich auf das Laufwerk bezieht, auf dem sich die Web Publishing Engine-Komponente Ihres FileMaker-Servers befindet.
   - **Für Apache (Mac OS):** `/Library/WebServer/Documents`

Hinweise

- Für Containerobjekte, die als Dateiverweise gespeichert sind, muss Ihr Web-Server so konfiguriert sein, dass er die MIME- (Multipurpose Internet Mail Extensions) Typen für alle Arten von Dateien unterstützt, die Sie anbieten wollen (z. B. Filme). Ihr Web-Server bestimmt die Unterstützung für die aktuellen MIME-Typen, die für das Internet registriert sind. Die Web Publishing Engine ändert die MIME-Unterstützung eines Web-Servers nicht. Weitere Informationen finden Sie in der Dokumentation zu Ihrem Web-Server.
- Alle QuickTime-Filme werden in Containerfeldern als Verweis gespeichert.

Containerfelder mit extern gespeicherten Daten

Wenn ein Containerfeld Objekte extern speichert — wenn Sie also **Containerdaten extern speichern** in dem FileMaker Pro-Dialogfeld „Feldoptionen“ ausgewählt haben — verwenden Sie den Assistenten für den Datenbank-Upload, um Datenbankdateien aus dem Client-Dateisystem zu FileMaker Server zu übertragen. Der Assistent für den Datenbank-Upload überträgt die Datenbank und die Containerfeld-Objekte in die für die Bereitstellung richtigen Ordner auf Ihrem Server. Weitere Informationen zur Verwendung des Assistenten für den Datenbank-Upload finden Sie in der FileMaker Server Hilfe. Weitere Informationen zum Einrichten von Containerfeldern für die externe Speicherung von Daten finden Sie in der FileMaker Pro Hilfe.

Wenn Sie eine Datenbank manuell hochladen, die ein Containerfeld mit extern gespeicherten Objekten verwendet, müssen Sie diese Schritte befolgen, um die extern gespeicherten Containerobjekte über die Web Publishing Engine zu veröffentlichen.

So laden Sie eine Datenbank manuell hoch:

1. Legen Sie die Datenbankdatei an den richtigen Ort auf dem Server. Legen Sie die FileMaker Pro-Datenbankdateien, die FileMaker Server öffnen soll – oder Verknüpfungen (Windows) bzw. Aliasdateien (MacOS) zu diesen Dateien – in den folgenden Ordnern ab:
   - **Windows (32-Bit):** `[Laufwerk]:\Programme\FileMaker\FileMaker Server\Data\Databases\`
   - **Windows (64-Bit):** `[Laufwerk]:\Programme (x86)\FileMaker\FileMaker Server\Data\Databases\`
   - **Mac OS:** `/Library/FileMaker Server/Data/Databases/`

   Sie können die Dateien auch in einem optional angegebenen, zusätzlichen Datenbankordner ablegen.
2. Erstellen Sie in dem Ordner, in den Sie die Datenbank gelegt haben, einen Ordner namens RC_Data_FMS, falls er nicht bereits existiert.

3. Erstellen Sie im Ordner RC_Data_FMS einen Ordner mit einem Namen, der dem Namen Ihrer Datenbank entspricht. Wenn Ihre Datenbank z. B. „Kunden“ heißt, erstellen Sie einen Ordner „Kunden“. Legen Sie die extern gespeicherten Objekte in den neu erstellten Ordner.

   Hinweis Wenn Datenbanken auf FileMaker Server bereitgestellt werden, gibt es keine Möglichkeit für verschiedene Datenbanken, auf einen gemeinsamen Ordner mit Containerobjekten zuzugreifen. Die Containerobjekte für jede Datenbank müssen sich in einem Ordner befinden, der durch den Namen der Datenbank identifiziert wird.

4. Für gemeinsam benutzte Dateien von Mac OS nehmen Sie die Dateien in die Gruppe **fmsadmin** auf.

   Weitere Informationen zum manuellen Hochladen von Datenbanken finden Sie in der FileMaker Server Hilfe.

Hinweis Die Web Publishing Engine unterstützt den progressiven Download von Audiodateien (.mp3), Videodateien (.mov, .mp4 und .avi empfohlen) und PDF-Dateien für interaktive Container. Zum Beispiel kann ein Web-Benutzer beginnen, einen Film anzusehen, auch wenn der Film noch nicht komplett heruntergeladen ist. Um den progressiven Download zuzulassen, müssen Sie die Dateien mithilfe von Optionen erstellen, die Streaming unterstützen oder für die Web-Anzeige optimiert wurden. Erstellen Sie beispielsweise Dateien mithilfe der Option „Für Web-Anzeige optimieren“.

Anzeige von Containerfelddaten durch Web-Benutzer

Bei der Veröffentlichung einer Datenbank mit der Web Publishing Engine gelten folgende Einschränkungen für die Containerfeldobjekte:

- Web-Benutzer können den Inhalt eines Containerfelds nicht ändern oder ihm Inhalte hinzufügen. Web-Benutzer können Containerfelder nicht verwenden, um Objekte in die Datenbank hochzuladen.
- Bei Datenbanken, die ein Containerfeld mit aktivierter Vorschau verwenden, lädt die Web Publishing Engine die gesamte Datei, keine Vorschau, herunter.

FileMaker-Scripts und Custom Web Publishing

Die Funktion „Scripts verwalten“ in FileMaker Pro kann sich häufig wiederholende Aufgaben automatisieren und mehrere Aufgaben zusammenfassen. In Kombination mit Custom Web Publishing ermöglichen FileMaker-Scripts den Web-Benutzern, mehrere Aufgaben oder eine Reihe von Aufgaben durchzuführen.

FileMaker unterstützt über 65 Scriptschritte in Custom Web Publishing. Web-Benutzer können eine Vielzahl automatisierter Aufgaben durchführen, wenn Sie Scripts in einer Query-Zeichenfolge für einen URL verwenden. Um nicht unterstützte Scriptschritte anzuzeigen, wählen Sie im Dialogfeld „Script bearbeiten“ in FileMaker Pro die Option **Custom Web Publishing** aus der Liste **Kompatibilität anzeigen**. Grau dargestellte Scripts werden nicht für Custom Web Publishing unterstützt. Informationen zum Erstellen von Scripts finden Sie in der FileMaker Pro Hilfe.

Tipps und Überlegungen zu Scripts

Auch wenn viele Scriptschritte im Web identisch funktionieren, gibt es einige, die sich anders verhalten. Weitere Informationen erhalten Sie unter „Scriptverhalten in Custom Web Publishing-Lösungen“ auf Seite 20. Testen Sie alle Scripts, die von einem Webbrowser ausgeführt werden, bevor Sie Ihre Datenbank bereitstellen. Stellen Sie sicher, dass Sie sich mit unterschiedlichen Benutzerkonten anmelden, damit Sie sichergehen können, dass die Scripts für alle Clients gleichermaßen funktionieren. Prüfen Sie die Protokolldatei der Web Publishing Engine (wpe.log) auf Scripting-bezogene Fehler. Weitere Informationen finden Sie unter „Verwenden des Web Publishing Engine-Protokolls“ auf Seite 44.

Beachten Sie folgende Tipps und Überlegungen:

- Verwenden Sie Konten und Zugriffsrechte, um die Scripts einzuschränken, die ein Web-Benutzer ausführen kann. Überprüfen Sie, ob die Scripts nur Web-kompatible Scriptschritte enthalten und nur Zugang zu Scripts gewähren, die von einem Webbrowser aus ausgeführt werden sollen.
- Berücksichtigen Sie die Nebeneffekte von Scripts, die eine Reihe von Scriptschritten ausführen, die durch Zugriffsrechte kontrolliert werden. Wenn ein Script beispielsweise einen Scriptschritt zum Löschen von Datensätzen enthält und der Web-Benutzer sich nicht mit einem Konto anmeldet, das das Löschen von Datensätzen zulässt, führt das Script den Scriptschritt zum Löschen von Datensätzen nicht aus. Das Script könnte jedoch weiter ausgeführt werden, so dass unerwartete Ergebnisse auftreten können.
- Wählen Sie im Dialogfeld „Script bearbeiten“ **Script mit vollen Zugriffsrechten ausführen**, damit Scripts Aufgaben ausführen können, für die Sie dem Benutzer an sich keinen Zugang gewähren wollen. Beispielsweise können Sie Benutzern untersagen, mit ihren Konten und Zugriffsrechten Datensätze zu löschen, ihnen aber die Ausführung eines Scripts gestatten, das unter vordefinierten Bedingungen innerhalb eines Scripts bestimmte Arten von Datensätzen löscht.
- Wenn Ihre Scripts Schritte enthalten, die nicht unterstützt werden (z. B. Schritte, die nicht Web-kompatibel sind), verwenden Sie den Scriptschritt **AnwenderAbbruchZulassen setzen**, um festzulegen, wie nachfolgende Schritte gehandhabt werden.
  - Wenn der Scriptschritt **AnwenderAbbruchZulassen setzen** aktiviert (ein) ist, hindern nicht unterstützte Scriptschritte das Script an der Fortsetzung.
  - Wenn **AnwenderAbbruchZulassen setzen** ausgeschaltet ist, werden nicht unterstützte Scriptschritte übersprungen und das Script setzt die Ausführung fort.
  - Wenn dieser Scriptschritt nicht enthalten ist, werden Scripts ausgeführt, als ob die Funktion aktiv wäre, d. h., nicht unterstützte Scriptschritte stoppen Scripts.
- Einige Scripts, die mit einem Schritt von einem FileMaker Pro-Client aus funktionieren, benötigen eventuell einen zusätzlichen Schritt **„Schreibe Änderung Datens./Abfrage“**, um die Daten auf dem Host zu speichern. Da Web-Benutzer nicht über eine direkte Verbindung zum Host verfügen, werden sie nicht sofort benachrichtigt, wenn sich Daten ändern. Funktionen wie bedingte Wertelisten funktionieren z. B. nicht gleichermaßen für Web-Benutzer, da die Daten auf dem Host gespeichert werden müssen, bevor sich die Wirkung in einem Wertelistenfeld zeigt.

- Scripts, die Daten verändern, sollten den Schritt „Schreibe Änderung Datens./Abfrage“ enthalten, da Datenänderungen erst im Browser sichtbar werden, wenn die Daten gespeichert bzw. am Server „bestätigt“ wurden. Dies gilt für einige Scriptschritte wie Ausschneiden, Kopieren, Einfügen usw. Viele Aktionen mit nur einem Scriptschritt sollten in Scripts umgewandelt werden, die den Schritt „Schreibe Änderung Datens./Abfrage“ enthalten. Wenn Sie Scripts entwerfen, die von einem Webbrowser aus ausgeführt werden, nehmen Sie den Schritt „Schreibe Änderung Datens./Abfrage“ am Ende des Scripts auf, um sicherzustellen, dass alle Änderungen gespeichert werden.
- Um von der Art des Clients abhängige Scripts zu erstellen, verwenden Sie die Funktion „Hole(ProgrammVersion)“. Wenn der zurückgegebene Wert „Web Publishing Engine“ enthält, wissen Sie, dass der aktuelle Benutzer über Custom Web Publishing auf Ihre Datenbank zugreift. Weitere Informationen zu Funktionen finden Sie in der FileMaker Pro Hilfe.

Scriptverhalten in Custom Web Publishing-Lösungen

Die nachfolgenden Scriptschritte arbeiten im Web anders als in FileMaker Pro. Weitere Informationen zu allen Scriptschritten finden Sie in der FileMaker Pro Hilfe.

Scriptschritt	Verhalten in Custom Web Publishing-Lösungen
Script ausführen	Scripts können nur dann in anderen Dateien ausgeführt werden, wenn die Dateien auf FileMaker Server bereitgestellt werden und Custom Web Publishing in den anderen Dateien aktiviert ist.
Programm beenden	Meldet Web-Benutzer ab, schließt die Fenster, beendet aber den Webbrowser nicht.
AnwenderAbbruchZulassen setzen	Legt fest, wie nicht unterstützte Scriptschritte gehandhabt werden. Aktivieren, damit Scripts gestoppt werden; deaktivieren, damit nicht unterstützte Schritte übersprungen werden. Weitere Informationen finden Sie unter „Tipps und Überlegungen zu Scripts“ auf Seite 19. Web-Benutzer können Custom Web Publishing-Scripts nicht abbrechen, aber diese Option ermöglicht, dass nicht unterstützte Scriptschritte die Fortsetzung des Scripts stoppen.
Fehleraufzeichnung setzen	Diese Option ist bei Custom Web Publishing immer aktiviert. Web-Benutzer können Custom Web Publishing-Scripts nicht abbrechen.
Scriptpause setzen	Dieses Script wird zwar in Custom Web Publishing unterstützt, sollte aber nicht benutzt werden. Wenn der Scriptschritt „Scriptpause setzen“ ausgeführt wird, hält das Script an. Nur ein Script, das erneut den Schritt „Scriptpause setzen“ enthält, kann die Ausführung fortsetzen. Wenn das Script im Pausezustand bleibt, bis die Sitzung endet, wird das Script nicht vollständig ausgeführt.
Sortieren	Sie müssen eine Sortierfolge mit dem Scriptschritt „Datensätze sortieren“ speichern, damit er in Custom Web Publishing ausgeführt wird.
URL öffnen	Dieser Scriptschritt hat keine Wirkung in einer Custom Web Publishing-Lösung.
Gehe zu Feld	Im Webbrowser können Sie „Gehe zu Feld“ nicht verwenden, um zu einem bestimmten Feld zu wechseln. Sie können diesen Scriptschritt jedoch in Kombination mit anderen Scriptschritten verwenden, um Aufgaben auszuführen. Beispielsweise können Sie zu einem Feld gehen und den Inhalt kopieren und dann zu einem anderen Feld gehen und den Wert einfügen. Um die Wirkung im Browser zu sehen, müssen Sie den Datensatz mit dem Scriptschritt „Datensatz bestätigen“ speichern.
Schreibe Änderung Datens./Abfrage	Sendet den Datensatz zur Datenbank.

Script-Trigger in Custom Web Publishing-Lösungen

In FileMaker Pro können sowohl Scripts als auch Benutzeraktionen (wie zum Beispiel ein Benutzer, der in ein Feld klickt) Script-Trigger aktivieren. In Custom Web Publishing können jedoch nur Scripts Script-Trigger aktivieren. Weitere Informationen zu Script-Triggern finden Sie in der FileMaker Pro Hilfe.

Hinweis In FileMaker Pro 12 hat sich das Dialogfeld „Dateioptionen“ geändert. Daraus resultierend müssen Sie den Script-Trigger „BeiErstemFensterÖffnen“ verwenden, wenn Sie ein Script ausführen möchten, wenn eine Datei geöffnet wird. Entsprechend müssen Sie den Script-Trigger „BeiLetztemFensterSchließen“ verwenden, wenn Sie ein Script ausführen möchten, wenn eine Datei geschlossen wird.

Kapitel 4
Zugriff auf XML-Daten mit der Web Publishing Engine

Sie können FileMaker-Daten in Extensible Markup Language- (XML) Format mithilfe der Web Publishing Engine beziehen und aktualisieren. So wie HTML zur Standardbeschreibungssprache für die Kommunikation im World Wide Web wurde, hat sich XML zur Standardsprache für den strukturierten Datenaustausch entwickelt. Diese Sprache wird von zahlreichen Einzelpersonen, Organisationen und Unternehmen zur Übermittlung von Produktinformationen, Transaktionen, Inventardaten und anderen geschäftlichen Informationen genutzt.

Verwenden von Custom Web Publishing mit XML

Wenn Sie Standard-XML kennen, können Sie mit dem Einsatz der Web Publishing Engine beginnen, nachdem Sie ein paar spezielle Details über Custom Web Publishing mit XML wie z. B. die erforderliche URL-Syntax und Query-Parameter gelernt haben.

Indem Sie HTTP-URL-Abfragen mit FileMaker-Query-Befehlen und -Parametern verwenden, können Sie eine Datenbank abfragen, die von FileMaker Server bereitgestellt wird, und die resultierenden Daten in XML-Format herunterladen. Sie können eine Datenbank beispielsweise auf alle Datensätze in einem bestimmten Postleitzahlenbereich abfragen und die resultierenden XML-Daten wie gewünscht verwenden.

Allgemeine Informationen zu XML, weitere Beispiele, in denen es verwendet wird, sowie Links zu XML-Ressourcen finden Sie auf der FileMaker-Website unter http.//www.filemaker.de.

Hinweis Die von der Web Publishing Engine generierten XML-Daten sind wohlgeformt und richten sich nach der XML 1.0-Spezifikation. Informationen zu den Anforderungen des XML-Aufbaus finden Sie in der XML-Spezifikation unter http://www.w3.org.

Unterschiede zwischen der Web Publishing Engine und FileMaker Pro XML-Import/Export

Die Web Publishing Engine und FileMaker Pro gestatten Ihnen, XML-Daten mit FileMaker-Datenbanken zu verwenden. Es gibt jedoch einige wesentliche Unterschiede zwischen den beiden Methoden:

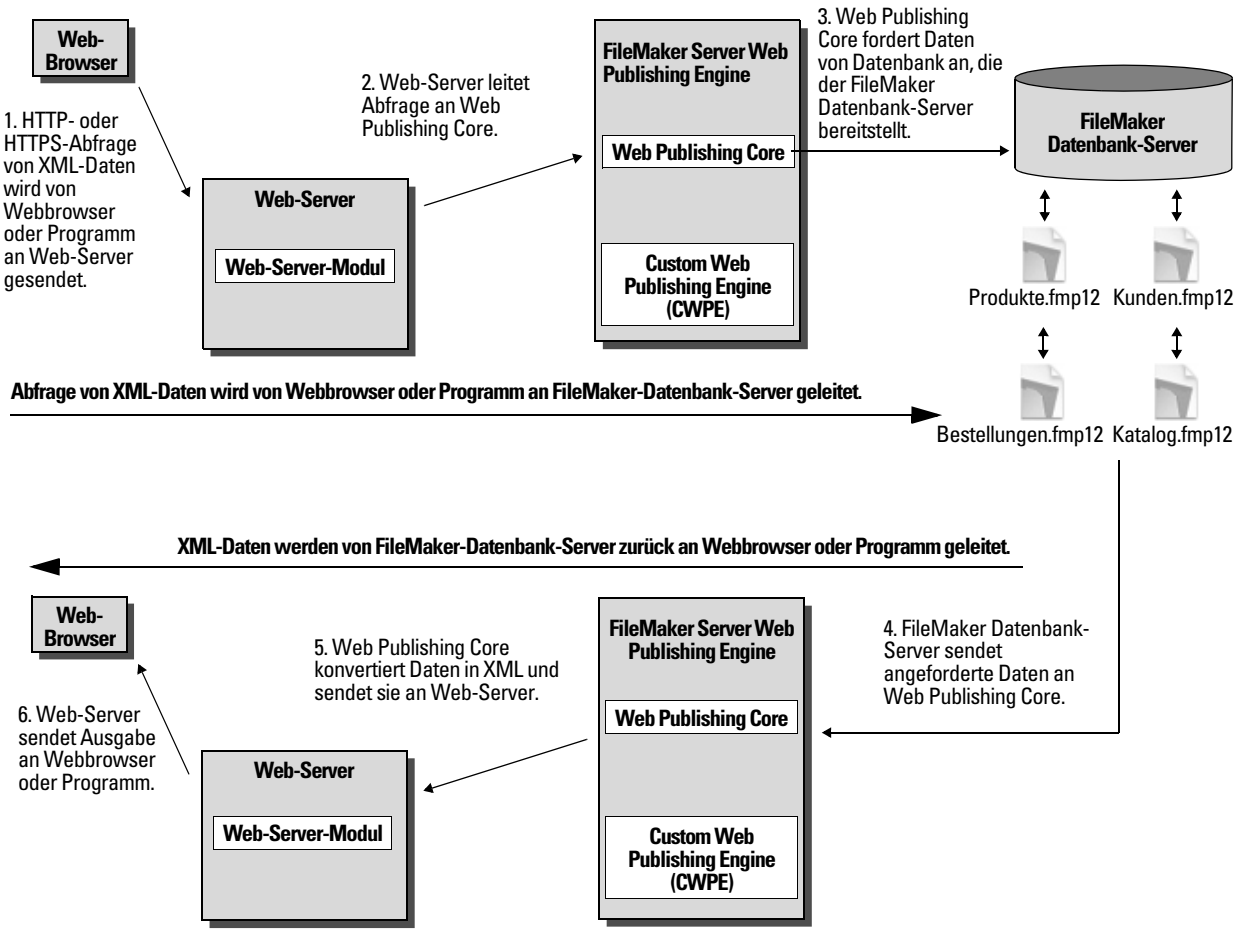
- Für den Zugriff auf XML-Daten unterstützt die Web Publishing Engine die `fmresultset`-, die `FMPXMLRESULT`- und die `FMPXMLLAYOUT`-Grammatik. Für XML-Import verwendet FileMaker Pro die `FMPXMLRESULT`-Grammatik und für den Export die `FMPXMLRESULT`-Grammatik. Weitere Informationen finden Sie unter „Zugriff auf XML-Daten über die Web Publishing Engine“ auf Seite 27.
- Um mit der Web Publishing Engine auf XML-Daten zuzugreifen, verwenden Sie einen Web Publishing Engine-Query-String in einem URL. Für den Import und Export von XML mit FileMaker Pro verwenden Sie FileMaker Pro-Menübefehle oder Scripts.
- Die Web Publishing Engine ist serverbasiert und kann auf demselben oder einem anderen Host als FileMaker Server installiert werden. FileMaker Pro XML-Import und -Export ist desktopbasiert.

- Sie können dynamisch auf XML-Daten von FileMaker-Datenbanken zugreifen, indem Sie URL-Abfragen mit der Web Publishing Engine verwenden. Die XML-Exportfunktion von FileMaker Pro erzeugt eine zuvor angegebene XML-Datendatei.
- Das Arbeiten mit XML-Daten über die Web Publishing Engine ist ein interaktiver Vorgang. FileMaker Pro XML-Import und -Export erfolgt im Stapelbetrieb.
- Die Web Publishing Engine kann auf XML-Daten von einem FileMaker-Ausschnitt zugreifen, FileMaker Pro nicht.
- Die Web Publishing Engine kann auf Daten in einem Containerfeld zugreifen, FileMaker Pro nicht.
- Die Web Publishing Engine bietet Echtzeitzugriff auf FileMaker-Daten über HTTP oder HTTPS, FileMaker Pro nicht.

Hinweis Informationen über das Verwenden von FileMaker Pro für Import und Export von Daten in XML-Format finden Sie in der FileMaker Pro Hilfe.

Wie die Web Publishing Engine aus einer Abfrage XML-Daten generiert

Nachdem eine Abfrage von XML-Daten an den Web-Server gesendet wurde, fragt die Web Publishing Engine die FileMaker-Datenbank ab und gibt die Daten als XML-Dokument zurück.


Web-Browser
1. HTTP- oder HTTPS-Abfrage von XML-Daten wird von Webbrowser oder Programm an Web-Server gesendet.
Web-Server
Web-Server-Modul
2. Web-Server leitet Abfrage an Web Publishing Core.
FileMaker Server Web Publishing Engine
Web Publishing Core
Custom Web Publishing Engine (CWPE)
3. Web Publishing Core fordert Daten von Datenbank an, die der FileMaker Datenbank-Server bereitstellt.
FileMaker Datenbank-Server
Produkte.fmp12
Kunden.fmp12
Bestellungen.fmp12
Katalog.fmp12
Abfrage von XML-Daten wird von Webbrowser oder Programm an FileMaker-Datenbank-Server geleitet.
XML-Daten werden von FileMaker-Datenbank-Server zurück an Webbrowser oder Programm geleitet.
Web-Browser
6. Web-Server sendet Ausgabe an Webbrowser oder Programm.
Web-Server
Web-Server-Modul
5. Web Publishing Core konvertiert Daten in XML und sendet sie an Web-Server.
FileMaker Server Web Publishing Engine
Web Publishing Core
Custom Web Publishing Engine (CWPE)
4. FileMaker Datenbank-Server sendet angeforderte Daten an Web Publishing Core.

Allgemeines Verfahren beim Zugriff auf XML-Daten von der Web Publishing Engine

Im Folgenden erhalten Sie einen Überblick über das Verfahren beim Verwenden der Web Publishing Engine für den Zugriff auf XML-Daten in einer FileMaker-Datenbank:

1. Stellen Sie in FileMaker Server Admin Console sicher, dass XML Publishing aktiviert ist. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

2. Öffnen Sie in FileMaker Pro jede FileMaker-Datenbank, die Sie veröffentlichen wollen, und stellen Sie sicher, dass in der Datenbank das erweiterte Zugriffsrecht **fmxml** für XML Custom Web Publishing aktiviert ist. Weitere Informationen finden Sie unter „Aktivieren von Instant Web Publishing in einer Datenbank“ auf Seite 14.

   Um auf XML-Daten in einem Ausschnitt zuzugreifen, stellen Sie die Ansicht für das Datenbanklayout auf **Als Formular anzeigen** oder **Als Liste anzeigen** ein. Wenn ein Benutzer oder Script die Ansicht des Datenbanklayouts zu **Als Tabelle anzeigen** ändert, kann nur auf den ersten Bezugsdatensatz (erste Zeile im Ausschnitt) als XML-Daten zugegriffen werden.

   Die XML-Daten werden in einer Reihenfolge ausgegeben, die der Reihenfolge entspricht, in der die Feldobjekte dem Layout hinzugefügt wurden. Wenn die XML-Datenreihenfolge der Reihenfolge entsprechen soll, in der die Felder auf dem Bildschirm angezeigt werden (von oben nach unten, von links nach rechts), dann wählen Sie alle Felder aus, gruppieren Sie sie und heben Sie dann die Gruppierung wieder auf. Dieser Vorgang setzt die Layoutreihenfolge wieder zurück auf die Bildschirmreihenfolge.

3. Senden Sie durch ein HTML-Formular, einen HREF-Link oder ein Script in Ihrem Programm oder Ihrer Web-Seite eine HTTP- oder HTTPS-Abfrage in Form eines URLs mit Angabe der FileMaker XML-Grammatik, einen Query-Befehl und einen oder mehrere FileMaker-Query-Parameter an die Web Publishing Engine. Der URL kann auch in einem Browser eingegeben werden.

   Informationen über die Angabe des URLs finden Sie im nächsten Abschnitt, „Erläuterung der URL-Syntax für XML-Daten und Containerobjekte“. Informationen über Query-Befehle und -Parameter finden Sie unter „Abfragen von XML-Daten mithilfe von FileMaker-Query-Strings“ auf Seite 37, und in Anhang A, „Gültige Namen in Query-Strings“.

4. Die Web Publishing Engine verwendet die im URL angegebene Grammatik, um XML-Daten mit den Ergebnissen Ihrer Abfrage zu erzeugen (z. B. eine Gruppe von Datensätzen aus der Datenbank), und gibt diese an Ihr Programm oder Ihren Webbrowser zurück.

5. Der Webbrowser, falls er über einen XML-Parser verfügt, zeigt die Daten an bzw. das Programm verwendet die Daten wie von Ihnen angegeben.

Erläuterung der URL-Syntax für XML-Daten und Containerobjekte

Dieser Abschnitt beschreibt die URL-Syntax beim Verwenden der Web Publishing Engine für den Zugriff auf XML-Daten und Containerobjekte in FileMaker-Datenbanken.

Erläuterung der URL-Syntax für XML-Daten

Die URL-Syntax beim Verwenden der Web Publishing Engine für den Zugriff auf XML-Daten in FileMaker-Datenbanken lautet:

```
<scheme>://<host>[:<port>]/fmi/xml/<xml_grammar>.xml[?<query string>]
```

Dabei gilt:

- `<scheme>` kann das HTTP- oder HTTPS-Protokoll sein.
- `<host>` ist die IP-Adresse oder der Domänenname des Hosts, auf dem der Web-Server installiert ist.
- `<port>` ist optional und gibt den Port an, den der Web-Server überwacht. Ohne Portangabe wird der Standard-Port des Protokolls (Port 80 für HTTP oder Port 443 für HTTPS) verwendet.
- `<xml_grammar>` ist der Name der FileMaker-XML-Grammatik. Mögliche Werte sind `fmresultset`, `FMPXMLRESULT` bzw. `FMPXMLLAYOUT`. Weitere Informationen hierzu finden Sie unter „Verwenden der fmresultset-Grammatik“ auf Seite 29 und „Verwenden anderer FileMaker XML-Grammatiken“ auf Seite 32.
- `<query string>` ist eine Kombination aus einem Query-Befehl und einem oder mehreren Query-Parametern für FileMaker XML Publishing. (Der Befehl `-dbnames` verlangt keine Parameter.) Weitere Informationen finden Sie unter „Abfragen von XML-Daten mithilfe von FileMaker-Query-Strings“ auf Seite 37, und Anhang A, „Gültige Namen in Query-Strings“.

Hinweis Die URL-Syntax einschließlich der Namen von Query-Befehl und -Parametern unterscheidet Groß- und Kleinschreibung, mit Ausnahme von Teilen des Query-Strings. Der URL ist vorwiegend in Kleinbuchstaben mit Ausnahme der zwei Grammatiknamen in Großbuchstaben: `FMPXMLRESULT` und `FMPXMLLAYOUT`. Informationen über die Regeln für Groß-/Kleinschreibung im Query-String finden Sie unter „Richtlinien für die Verwendung von Query-Befehlen und -Parametern“ auf Seite 47.

Zwei URL-Beispiele für den Zugriff auf XML-Daten über die Web Publishing Engine:

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=products&-lay=sales
&-findall
```

```
http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?-db=products&-lay=sales
&-findall
```

Erläuterung der URL-Syntax für FileMaker-Containerobjekte in XML-Lösungen

In einem generierten XML-Dokument für eine XML-Lösung unterscheidet sich die Syntax für den Verweis auf ein Containerobjekt für Containerfelder, die das tatsächliche Objekt in der Datenbank speichern, und für Containerfelder, die einen Verweis auf das Objekt speichern.

Wenn ein Containerfeld das tatsächliche Objekt in der Datenbank speichert

Das `<data>`-Element des Containerfelds verwendet die folgende relative URL-Syntax, um sich auf das Objekt zu beziehen:

```
<data>/fmi/xml/cnt/data.<extension>?<query string></data>
```

Dabei gilt: `<extension>` ist die Dateinamenerweiterung zur Identifizierung des Objekttyps wie beispielsweise .jpg. Die Dateinamenerweiterung stellt den MIME-Typ ein, damit der Webbrowser die Containerdaten korrekt identifizieren kann. Informationen über `<query string>` finden Sie im vorherigen Abschnitt, „Erläuterung der URL-Syntax für XML-Daten“.

Beispiel:

```
<data>/fmi/xml/cnt/data.jpg?-db=products&-lay=sales&-field=product_image(1)
&-recid=2</data>
```

Hinweis In der generierten XML für ein Containerfeld ist der Wert für den Query-Parameter `-field` ein vollständig qualifizierter Feldname. Die Zahl in Klammern gibt die Wiederholungsnummer für das Containerfeld an und wird für Wiederhol- und für Einzelfelder generiert. Weitere Informationen finden Sie unter „Erläuterung der Syntax für einen voll qualifizierten Feldnamen“ auf Seite 49.

Verwenden Sie zum Abrufen der Containerdaten aus der Datenbank die folgende Syntax:

```
<scheme>://<host>[:<port>]/fmi/xml/cnt/data.<extension>?<query string>
```

Informationen über `<schema>`, `<host>` oder `<port>` finden Sie im vorherigen Abschnitt, „Erläuterung der URL-Syntax für XML-Daten“.

Beispiel:

```
http://www.company.com/fmi/xml/cnt/data.jpg?-db=products&-lay=sales
&-field=product_image(1)&-recid=2
```

Wenn ein Containerfeld einen Dateiverweis anstelle eines tatsächlichen Objekts speichert

Das `<data>`-Element des Containerfelds enthält einen relativen Pfad, der auf das Objekt verweist. Beispiel:

```
<data>/images/logo.jpg</data>
```

Hinweis Das Containerobjekt, auf das verwiesen wird, muss im Ordner „FileMaker Pro Web“ gespeichert sein, wenn der Datensatz erstellt oder bearbeitet wird, und dann in einen Ordner mit dem gleichen relativen Speicherort im Root-Ordner der Web-Server-Software kopiert oder verschoben werden. Weitere Informationen finden Sie unter „Veröffentlichen des Inhalts von Containerfeldern im Web“ auf Seite 16.

Wenn ein Containerfeld leer ist

Das `<data>`-Element des Containerfelds ist leer.

Erläuterung von URL-Textkodierung

Die URLs für den Zugriff auf XML-Daten und Containerfelder müssen in UTF-8 (Unicode Transformation 8 Bit)-Format kodiert werden. Informationen finden Sie unter „Erläuterung von UTF-8-kodierten Daten“ auf Seite 37.

Um z. B. den Wert des „Info“-Felds in `Matinée` zu ändern, könnten Sie folgenden URL verwenden:

```
http://server.company.com/fmi/xml/fmresultset.xml?-db=members
&-lay=relationships&-recid=2&info= matin%C3%A9e&-edit
```

In diesem Beispiel-URL ist `%C3%A9` die URL-kodierte UTF-8-Darstellung des Zeichens `é`.

Weitere Informationen zur URL-Textkodierung finden Sie in der URL-Spezifikation unter http://www.w3.org.

Zugriff auf XML-Daten über die Web Publishing Engine

Um über die Web Publishing Engine auf XML-Daten zuzugreifen, verwenden Sie einen URL, der den Namen der zu verwendenden FileMaker-Grammatik, einen FileMaker-Query-Befehl und einen oder mehrere FileMaker-Query-Parameter angibt. Die Web Publishing Engine generiert XML-Daten aus Ihrer Datenbank, die nach einem der folgenden Typen von XML-Grammatiken formatiert werden:

- **fmresultset:** Dies ist die empfohlene Grammatik für die Web Publishing Engine für den Zugriff auf XML-Daten. Sie ist flexibel und für bequemen Feldzugriff nach Name und einfacherer Änderung von `relatedset-` (Ausschnitt-) Daten optimiert. Diese Grammatik ist auch direkter mit FileMaker-Terminologie verbunden und bietet z. B. globale Speicheroptionen und Identifikation von Statistik- und Formelfeldern. Für einfacheres Web Publishing wurde diese Grammatik knapper formuliert als die `FMPXMLRESULT`-Grammatik. Weitere Informationen finden Sie unter „Verwenden der fmresultset-Grammatik“ auf Seite 29.
- **FMPXMLRESULT und FMPXMLLAYOUT:** Sie können auch die `FMPXMLRESULT`- und `FMPXMLLAYOUT`-Grammatiken mit der Web Publishing Engine für den Zugriff auf XML-Daten verwenden. Um ein einziges Stylesheet sowohl für XML-Export als auch für Custom Web Publishing einzusetzen, müssen Sie die `FMPXMLRESULT`-Grammatik benutzen. Für den Zugriff auf Wertelisten und Feldanzeigedaten in Layouts müssen Sie die `FMPXMLLAYOUT`-Grammatik verwenden. Weitere Informationen finden Sie unter „Verwenden anderer FileMaker XML-Grammatiken“ auf Seite 32.

Abhängig von der Grammatik, die Sie in der URL-Abfrage angegeben haben, generiert die Web Publishing Engine mit einer der folgenden Grammatiken ein XML-Dokument. Jedes XML-Dokument enthält eine XML-Standard-Namespace-Deklaration für die Grammatik. Informationen finden Sie im nächsten Abschnitt, „Erläuterung von Namensräumen für FileMaker XML“. Verwenden Sie eine dieser Grammatiken in Ihrem Dokument oder auf Ihrer Web-Seite, um FileMaker-Daten im XML-Format anzuzeigen und zu bearbeiten.

Hinweis Von der Web Publishing Engine generierte XML-Daten werden im UTF-8-Format (Unicode Transformation Format 8) kodiert. Weitere Informationen finden Sie unter „Erläuterung von UTF-8-kodierten Daten“ auf Seite 37.

Erläuterung von Namensräumen für FileMaker XML

Mithilfe eindeutiger XML-Namensräume können Sie XML-Tags der richtigen Anwendung zuordnen. Wenn Ihr XML-Dokument beispielsweise zwei `<DATABASE>`-Elemente enthält, eines für FileMaker-XML-Daten, das andere für Oracle-XML-Daten, kann anhand der Namensräume festgestellt werden, welches `<DATABASE>`-Element zu welchen XML-Daten gehört.

Die Web Publishing Engine generiert für jede Grammatik einen Standard-Namespace.

Grammatik	Generierter Standard-Namespace
`fmresultset`	`xmlns="http://www.filemaker.com/xml/fmresultset"`
`FMPXMLRESULT`	`xmlns="http://www.filemaker.com/ fmpxmlresult"`
`FMPXMLLAYOUT`	`xmlns="http://www.filemaker.com/fmpxmllayout"`

Erläuterung der FileMaker-Datenbankfehlercodes

Die Web Publishing Engine gibt in den „error code“-Elementen am Beginn jedes XML-Dokuments einen Fehlercode für einen etwaigen Fehler zurück, der im zuletzt ausgeführten Query-Befehl aufgetreten ist. Null (0) bedeutet keinen Fehler.

Grammatik	Verwendete Syntax
fmresultset	<error code="0"></error>
FMPXMLRESULT	<ERRORCODE>0</ERRORCODE>
FMPXMLLAYOUT	<ERRORCODE>0</ERRORCODE>

Das „error code“-Element im XML-Dokument gibt Fehler an, die die Datenbank und Query-Strings betreffen. Weitere Informationen finden Sie unter Anhang B, „Fehlercodes für Custom Web Publishing“.

Abrufen der Document Type Definitions für die FileMaker-Grammatiken

Sie können die Document Type Definitions (DTDs) für die FileMaker-Grammatiken mithilfe einer HTTP-Abfrage abrufen.

Grammatik	HTTP-Abfrage
fmresultset	http://<host>[:<port]/fmi/xml/fmresultset.dtd
FMPXMLRESULT	http://<host>[:<port]/fmi/xml/FMPXMLRESULT.dtd
FMPXMLLAYOUT	http://<Host>[:<Port]/fmi/xml/FMPXMLLAYOUT.dtd

Verwenden der fmresultset-Grammatik

Die XML-Elementnamen in dieser Grammatik verwenden FileMaker-Terminologie und die Speicherung von Feldern erfolgt getrennt von den Feldtypen. Die Grammatik umfasst auch die Fähigkeit, Statistik-, Formel- und Variablenfelder zu identifizieren.

Geben Sie zur Verwendung der `fmresultset`-Grammatik den folgenden Namen der `fmresultset`-Grammatik im URL an, der das XML-Dokument von der Web Publishing Engine anfordert:

```
fmresultset.xml
```

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-findall
```

Hinweis Geben Sie die `fmresultset`-Grammatik unbedingt in Kleinbuchstaben an.

Die Web Publishing Engine generiert ein XML-Dokument mithilfe der `fmresultset`-Grammatik. Im XML-Dokument verweist die Web Publishing Engine auf die Document Type Definition für die `fmresultset`-Grammatik in der Anweisung `<!DOCTYPE>` in der zweiten Zeile des Dokuments, unmittelbar nach der Anweisung `<?xml . . . ?>`. Die Anweisung `<!DOCTYPE>` gibt den URL für das Herunterladen der DTD für die `fmresultset`-Grammatik an.

Beschreibung der Elemente der fmresultset-Grammatik

Die `fmresultset`-Grammatik besteht hauptsächlich aus dem Element `<datasource>`, dem Element `<metadata>` und dem Element `<resultset>`.

Element <datasource>

In der `fmresultset`-Grammatik enthält das `<datasource>`-Element die Attribute „table“, „layout“, „date-format“, „time-format“, „timestamp-format“, „total-count“ und „database“.

- Das Attribut „date-format“ des `<datasource>`-Elements gibt das Format von Datumsangaben im XML-Dokument an:

  ```
  MM.tt.jjjj
  ```

Dabei gilt:

- `MM` ist der zweistellige Wert für den Monat (01 bis 12, wobei 01 für Januar und 12 für Dezember steht).
- `tt` ist der zweistellige Wert für den Tag des Monats (01 bis 31).
- `jjjj` ist der vierstellige Wert für das Jahr.

- Das Attribut „time-format“ des `<datasource>`-Elements gibt das Format von Zeitangaben im XML-Dokument an.

  ```
  HH:mm:ss
  ```

Dabei gilt:

- `HH` ist der zweistellige Wert für Stunden (00 bis 23 im 24-Stunden-Format).
- `mm` ist der zweistellige Wert für Minuten (00 bis 59).
- `ss` ist der zweistellige Wert für Sekunden (00 bis 59).

- Das Attribut „timestamp-format“ des `<datasource>`-Elements gibt das Format von Datums- und Zeitangaben in einem Zeitstempel an.

  ```
  MM.tt.jjjj HH:mm:ss
  ```

Element <metadata>

Das `<metadata>`-Element der `fmresultset`-Grammatik enthält ein oder mehrere `<field-definition>`- und `<relatedset-definition>`-Elemente, jeweils mit Attributen für eines der Felder in der Ergebnismenge.

Die Attribute `<field-definition>` geben an:

- ob das Feld ein Feld mit `automatischer Eingabe` ist („yes“ oder „no“)
- ob das Feld ein Feld mit `vierstelligem Jahr` ist („yes“ oder „no“)
- ob das Feld ein globales Feld ist („yes“ oder „no“)
- die maximale Anzahl an Wiederholwerten (Attribut `max-repeat`)
- die maximale Anzahl an zulässigen Zeichen (Attribut `max-characters`)
- ob das Feld ein `nicht leeres` Feld ist („yes“ oder „no“)
- ob das Feld ein Feld nur mit numerischen Daten ist („yes“ oder „no“)
- result („text“, „number“, „date“, „time“, „timestamp“ oder „container“)
- ob das Feld ein `Uhrzeit-des-Tages`-Feld ist („yes“ oder „no“)
- type („normal“, „calculation“ oder „summary“)
- den Feldnamen (bei Bedarf vollständig qualifiziert)

Das `<relatedset-definition>`-Element repräsentiert einen Ausschnitt. Jedes Bezugsfeld in einem Ausschnitt wird durch das `<field-definition>`-Element innerhalb des `<relatedset-definition>`-Elements dargestellt. Wenn ein Ausschnitt mehrere Bezugsfelder enthält, werden die Felddefinitionen für die Bezugsfelder in einem einzigen `<relatedset-definition>`-Element gruppiert.

Element <resultset>

Das `<resultset>`-Element enthält alle `<record>`-Elemente, die als Ergebnis einer Query zurückgegeben wurden, sowie ein Attribut für die Gesamtzahl der gefundenen Datensätze. Jedes `<record>`-Element enthält die Felddaten für einen Datensatz in der Ergebnismenge – einschließlich der Attribute `mod-id` und `record-id` für den Datensatz – sowie das `<data>`-Element mit den Daten für ein Feld im Datensatz.

Jeder Datensatz in einem Ausschnitt wird durch ein `<record>`-Element innerhalb des `<relatedset>`-Elements dargestellt. Das Attribut „count“ des `<relatedset>`-Elements gibt die Anzahl der Datensätze in einem Ausschnitt an und das Attribut „table“ gibt die mit dem Ausschnitt verbundene Tabelle an.

Beispiel für XML-Daten in der fmresultset-Grammatik

Nachfolgend erhalten Sie ein Beispiel für XML-Daten, die mit der `fmresultset`-Grammatik generiert wurden.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN"
""http://localhost:80/fmi/xml/fmresultset.dtd">
  <fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
  <error code="0" />
  <product build="12/31/2012" name="FileMaker Web Publishing Engine"
version="0.0.0.0" />
  <datasource database="art" date-format="MM/dd/yyyy" layout="web3" table="art"
time-format="HH:mm:ss" timestamp-format="MM/dd/yyyy HH:mm:ss" total-count="12"
/>
  <metadata>
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="Title" not-empty="no" numeric-only="no" result="text" time-of-
day="no" type="normal" />
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="Artist" not-empty="no" numeric-only="no" result="text" time-
of-day="no" type="normal" />
  <relatedset-definition table="artlocations">
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="artlocations::Location" not-empty="no" numeric-only="no"
result="text" time-of-day="no" type="normal" />
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="artlocations::Date" not-empty="no" numeric-only="no"
result="date" time-of-day="no" type="normal" />
  </relatedset-definition>
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="Style" not-empty="no" numeric-only="no" result="text" time-of-
day="no" type="normal" />
  <field-definition auto-enter="no" four-digit-year="no" global="no" max-
repeat="1" name="Length" not-empty="no" numeric-only="no" result="number" time-
of-day="no" type="calculation" />
  </metadata>
  <resultset count="1" fetch-size="1">
  <record mod-id="6" record-id="14">
  <field name="Title">
  <data>Spring in Giverny 3</data>
  </field>
  <field name="Artist">
  <data>Claude Monet</data>
  </field>
  <relatedset count="0" table="artlocations" />
  <field name="Style">
  <data />
  </field>
  <field name="length">
```

```
<data>19</data>
</field>
</record>
</resultset>
</fmresultset>
```

Verwenden anderer FileMaker XML-Grammatiken

Die anderen FileMaker XML-Grammatiken enthalten Informationen über Feldtypen, Wertelisten und Layouts. FMPXMLRESULT und fmresultset sind hinsichtlich der Funktionalität äquivalent. Für den Zugriff auf Wertelisten und Feldanzeigedaten in Layouts müssen Sie die FMPXMLLAYOUT-Grammatik verwenden. Die FMPXMLRESULT- und FMPXMLLAYOUT-Grammatiken sind kompakter für den Datenaustausch.

Geben Sie zur Verwendung der FMPXMLRESULT-Grammatik den folgenden Grammatiknamen im URL an, der das XML-Dokument von der Web Publishing Engine anfordert:

```
FMPXMLRESULT.xml
```

Beispiel:

```
http://192.168.123.101/fmi/xml/FMPXMLRESULT.xml?-db=employees&-lay=family
&-findall
```

Geben Sie zur Verwendung der FMPXMLLAYOUT-Grammatik den folgenden Grammatiknamen mit dem Query-Befehl -view im URL an, der das XML-Dokument von der Web Publishing Engine anfordert:

```
FMPXMLLAYOUT.xml
```

Beispiel:

```
http://192.168.123.101/fmi/xml/FMPXMLLAYOUT.xml?-db=employees&-lay=family
&-view
```

Hinweis Geben Sie die FMPXMLRESULT- und die FMPXMLLAYOUT-Grammatik unbedingt in Großbuchstaben an.

Im generierten XML-Dokument verweist die Web Publishing Engine auf die Document Type Definition für die fmresultset-Grammatik in der Anweisung <!DOCTYPE> in der zweiten Zeile des Dokuments, unmittelbar nach der Anweisung <?xml . . . ?>. Die Anweisung <!DOCTYPE> gibt den URL für das Herunterladen der DTD für die Grammatik an.

Beschreibung der Elemente in der FMPXMLRESULT-Grammatik

In der FMPXMLRESULT-Grammatik enthält das <DATABASE>-Element die Attribute NAME, RECORDS, DATEFORMAT, LAYOUT und TIMEFORMAT.

Das Attribut DATEFORMAT des <DATABASE>-Elements gibt das Datumsformat im XML-Dokument an. Das Attribut TIMEFORMAT des <DATABASE>-Elements gibt das Zeitformat im XML-Dokument an. Datums- und Zeitformate der Grammatiken FMPXMLRESULT und fmresultset sind gleich (siehe dazu die Tabellen unter „Beschreibung der Elemente der fmresultset-Grammatik“ auf Seite 29.)

Das <METADATA>-Element der FMPXMLRESULT-Grammatik enthält ein oder mehrere <FIELD>-Elemente. Jedes FIELD-Element enthält Daten für ein Feld bzw. eine Spalte in der Ergebnismenge einschließlich des Feldnamens, wie er in der Datenbank definiert ist, den Feldtyp, die Ja/Nein-Zulässigkeit für leere Felder (Attribut EMPTYOK) und die maximale Anzahl von Wiederholwerten (Attribut MAXREPEAT). Gültige Werte für Feldtypen sind TEXT, NUMBER, DATE, TIME, TIMESTAMP und CONTAINER.

Das <RESULTSET>-Element enthält alle <ROW>-Elemente, die als Ergebnis einer Query zurückgegeben wurden, sowie ein Attribut für die Gesamtzahl der gefundenen Datensätze. Jedes <ROW>-Element enthält die Feld-/Spalten-Daten für jede Zeile in der Ergebnismenge. Diese Daten umfassen RECORDID und MODID für die Zeile (siehe „Query-Parameter –modid (Änderungs-ID)“ auf Seite 59) und das <COL>-Element. Das <COL>-Element enthält die Daten für ein Feld/eine Spalte in der Zeile, wobei mehrere <DATA>-Elemente einen der Werte in einem Wiederhol- oder Ausschnittfeld repräsentieren.

Beispiel für XML-Daten in der FMPXMLRESULT-Grammatik

Nachfolgend sehen Sie ein Beispiel für XML-Daten, die mit der FMPXMLRESULT-Grammatik generiert wurden.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN"
""http://localhost:80/fmi/xml/FMPXMLRESULT.dtd">
<FMPXMLRESULT xmlns="http://www.filemaker.com/fmpxmlresult">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="31.12.2012" NAME="FileMaker Web Publishing Engine"
VERSION="0.0.0.0" />
  <DATABASE DATEFORMAT="MM/dd/yyyy" LAYOUT="web" NAME="art" RECORDS="12"
TIMEFORMAT="HH:mm:ss" />
  <METADATA>
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Title" TYPE="TEXT" />
     <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Artist" TYPE="TEXT" />
    <FIELD EMPTYOK="YES" MAXREPEAT="1" NAME="Image" TYPE="CONTAINER" />
  </METADATA>
  <RESULTSET FOUND="1">
  <ROW MODID="6" RECORDID="15">
  <COL>
    <DATA>Spring in Giverny 4</DATA>
  </COL>
  <COL>
    <DATA>Claude Monet</DATA>
  </COL>
  <COL>
    <DATA>/fmi/xml/cnt/data.jpg?-db=art&-lay=web&-recid=15&-
field=Image(1)</DATA>
  </COL>
  </ROW>
  </RESULTSET>
</FMPXMLRESULT>
```

Die Reihenfolge der <COL>-Elemente entspricht der Reihenfolge der <FIELD>-Elemente im <METADATA>-Element – z. B. entspricht die Reihenfolge der Felder „Title“ und „Artist“ im <METADATA>-Element der Reihenfolge von „Village Market“ und „Camille Pissarro“ in den Elementen <RESULTSET> und <ROW>.

Beschreibung der Elemente der Grammatik FMPXMLLAYOUT

In der FMPXMLLAYOUT-Grammatik enthält das <LAYOUT>-Element den Namen des Layouts, den Namen der Datenbank sowie <FIELD>-Elemente für jedes Feld im entsprechenden Layout der Datenbank. Jedes <FIELD>-Element beschreibt den Feldtyp und enthält das Attribut VALUELIST für jede Werteliste des Feldes.

Das <VALUELISTS>-Element enthält ein oder mehrere <VALUELIST>-Elemente für jede Werteliste im Layout – jedes VALUELIST-Element enthält den Namen der Werteliste und ein <VALUE>-Element für jeden Wert in der Liste.

Abhängig von den im Dialogfeld **Felder für Werteliste angeben** in der FileMaker-Datenbank gewählten Optionen enthält das Element <VALUE> ein DISPLAY-Attribut, das den Wert nur im ersten Feld, im zweiten Feld oder in beiden Feldern einer Werteliste enthält. Nehmen Sie zum Beispiel an, dass das erste Feld in einer Werteliste die ID-Nummer des Kunststils (wie z. B. „100“) und das zweite Feld den zugehörigen Namen des Kunststils (wie z. B. „Impressionismus“) enthält. Hier finden Sie eine Zusammenfassung des Inhalts des Attributs DISPLAY, wenn die verschiedenen Kombinationen der Optionen im Dialogfeld **Felder für Werteliste angeben** ausgewählt sind:

- Wenn **Auch Werte aus zweitem Feld anzeigen** nicht ausgewählt ist, enthält das Attribut DISPLAY nur den Wert im ersten Feld einer Werteliste. Im folgenden XML-Datenbeispiel enthält das Attribut DISPLAY nur die ID-Nummer des Kunststils:

```
<VALUELISTS>
      <VALUELIST NAME="style">
        <VALUE DISPLAY="100">100</VALUE>
        <VALUE DISPLAY="101">101</VALUE>
        <VALUE DISPLAY="102">102</VALUE>
      </VALUELIST>
</VALUELISTS>
```

- Wenn sowohl **Auch Werte aus zweitem Feld anzeigen** als auch **Werte nur aus dem zweiten Feld anzeigen** ausgewählt ist, enthält das DISPLAY-Attribut nur den Wert im zweiten Feld. Im folgenden XML-Datenbeispiel enthält das Attribut DISPLAY nur den Namen des Kunststils:

```
<VALUELISTS>
      <VALUELIST NAME="style">
        <VALUE DISPLAY="Impressionism">100</VALUE>
        <VALUE DISPLAY="Cubism">101</VALUE>
        <VALUE DISPLAY="Abstract">102</VALUE>
      </VALUELIST>
</VALUELISTS>
```

- Wenn **Auch Werte aus zweitem Feld anzeigen** ausgewählt ist und **Werte nur aus dem zweiten Feld anzeigen** nicht ausgewählt ist, enthält das DISPLAY-Attribut die Werte in beiden Feldern einer Werteliste. Im folgenden XML-Datenbeispiel enthält das Attribut DISPLAY sowohl die ID-Nummer als auch den Namen des Kunststils:

```
<VALUELISTS>
      <VALUELIST NAME="style">
        <VALUE DISPLAY="100 Impressionism">100</VALUE>
        <VALUE DISPLAY="101 Cubism">101</VALUE>
        <VALUE DISPLAY="102 Abstract">102</VALUE>
      </VALUELIST>
</VALUELISTS>
```

Für Datums-, Zeit- und Zeitstempelfelder werden Daten für Wertelisten über das „fm“-Format für diesen Feldtyp formatiert. Die „fm“-Formate sind `MM/tt/jjjj` für Datum, `HH:mm:ss` für Zeit und `MM/tt/jjjj HH:mm:ss` für Zeitstempel. Wenn zum Beispiel eine Werteliste „Geburtstage“ für ein Einblendmenü für ein Feld „Geburtstag“ eines Layouts verwendet wird und das Feld „Geburtstag“ ein Datumsfeld ist, werden alle für diese Werteliste ausgegebenen Werte im Datumsformat „fm“ ausgegeben.

Hinweis Wenn zwei Felder mit unterschiedlichem Feldtyp in einem Layout die gleiche Werteliste verwenden, bestimmt das erste Feld das Format der Daten der Werteliste.

XML-Daten in der FMPXMLLAYOUT-Grammatik – Beispiel

Im Folgenden finden Sie ein Beispiel für XML-Daten, die mit der `FMPXMLLAYOUT`-Grammatik generiert wurden.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLLAYOUT PUBLIC "-//FMI//DTD FMPXMLLAYOUT//EN"
""http://localhost:80/fmi/xml/FMPXMLLAYOUT.dtd">
  <FMPXMLLAYOUT xmlns="http://www.filemaker.com/fmpxmllayout">
  <ERRORCODE>0</ERRORCODE>
  <PRODUCT BUILD="31.12.2012" NAME="FileMaker Web Publishing Engine"
VERSION="0.0.0.0" />
  <LAYOUT DATABASE="art" NAME="web2">
  <FIELD NAME="Title">
    <STYLE TYPE="EDITTEXT" VALUELIST="" />
  </FIELD>
  <FIELD NAME="Artist">
    <STYLE TYPE="EDITTEXT" VALUELIST="" />
  </FIELD>
  <FIELD NAME="Image">
    <STYLE TYPE="EDITTEXT" VALUELIST="" />
  </FIELD>
  <FIELD NAME="artlocations::Location">
    <STYLE TYPE="EDITTEXT" VALUELIST="" />
  </FIELD>
  <FIELD NAME="artlocations::Date">
    <STYLE TYPE="EDITTEXT" VALUELIST="" />
  </FIELD>
  <FIELD NAME="Style">
    <STYLE TYPE="POPUPMENU" VALUELIST="style"/>
  </FIELD>
  </LAYOUT>
    <VALUELISTS>
      <VALUELIST NAME="style">
        <VALUE DISPLAY="Impressionism">100</VALUE>
        <VALUE DISPLAY="Cubism">101</VALUE>
        <VALUE DISPLAY="Abstract">102</VALUE>
      </VALUELIST>
    </VALUELISTS>
  </FMPXMLLAYOUT>
```

Erläuterung von UTF-8-kodierten Daten

Alle von der Web Publishing Engine generierten XML-Daten werden im UTF-8-Format (Unicode Transformation Format 8) kodiert. Dieses Format komprimiert ASCII-Zeichen im Unicode-Standardformat von 16 Bit auf 8 Bit. Für Unicode und UFT-8 sind XML-Parser erforderlich.

UTF-8-Kodierung unterstützt die direkte Darstellung der Werte 0-127 für den ASCII-Standardzeichensatz in der englischen Sprache und bietet Multibyte-Kodierungen für Unicode-Zeichen mit höheren Werten.

Hinweis Stellen Sie sicher, dass Sie einen Webbrowser oder ein Textverarbeitungsprogramm verwenden, das UTF-8-Dateien unterstützt.

Das Kodierungsformat UTF-8 hat die folgenden Eigenschaften:

- Alle ASCII-Zeichen sind UTF-8-Zeichen von der Größe eines Byte. Eine gültige ASCII-Zeichenkette ist eine gültige UTF-8-Zeichenkette.
- Jedes Nicht-ASCII-Zeichen (beliebiges Zeichen mit gesetztem höherwertigem Bit) ist Teil eines Multibyte-Zeichens.
- Das erste Byte eines UTF-8-Zeichens gibt die Anzahl der zusätzlichen Bytes im Zeichen an.
- Das erste Byte eines Multibyte-Zeichens lässt sich leicht vom darauf folgenden Byte unterscheiden, wodurch sich der Beginn eines Zeichens von einer beliebigen Position in einem Datenstrom einfach ermitteln lässt.
- Eine Konvertierung zwischen UTF-8 und Unicode ist einfach vorzunehmen.
- Die UTF-8-Kodierung ist relativ kompakt. Bei Text, der einen hohen Prozentsatz an ASCII-Zeichen enthält, ist sie kompakter als Unicode. Im schlimmsten Fall ist ein UTF-8-String nur 50 % größer als der entsprechende Unicode-String.

Abfragen von XML-Daten mithilfe von FileMaker-Query-Strings

Um XML-Daten aus einer FileMaker-Datenbank abzufragen, verwenden Sie die FileMaker-Query-Befehle und -Parameter in einem Query-String. Sie können beispielsweise den Query-Befehl `-findall` im folgenden Query-String in einem URL verwenden, um eine Liste aller Produkte aus einer FileMaker-Datenbank mit dem Namen „products“ anzufordern:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=products-lay=sales&-findall
```

Ein Query-String darf nur einen Query-Befehl, z. B. `-new`, enthalten. Die meisten Query-Befehle verlangen auch mehrere entsprechende Query-Parameter im Query-String. Beispielsweise verlangen alle Query-Befehle außer `-dbnames` den Parameter `-db`, der die abzufragende Datenbank angibt.

Sie können Query-Befehle und -Parameter auch in einem URL verwenden.

Dieser Abschnitt bietet eine Zusammenfassung der FileMaker Query-Befehle und -Parameter. Weitere Informationen über ihre Verwendung in einem Query-String finden Sie in „Gültige Namen in Query-Strings“ auf Seite 47.

Verwenden Sie diesen Query-Befehlsnamen	Zur Ausführung dieses Befehls
`-dbnames`	Namen aller bereitgestellten und im Web gemeinsam genutzten Datenbanken abrufen.
`-delete`	Datensatz löschen.

Verwenden Sie diesen Query-Befehlsnamen	Zur Ausführung dieses Befehls
`-dup`	Datensatz duplizieren.
`-edit`	Datensatz bearbeiten.
`-find`	Datensatz/Datensätze suchen.
`-findall`	Nach allen Datensätzen suchen.
`-findany`	Zufällig ermittelten Datensatz anzeigen.
`-findquery`	Komplexe oder zusammengesetzte Suchabfrage durchführen.
`-layoutnames`	Namen aller verfügbaren Layouts für eine bereitgestellte und im Web gemeinsam genutzte Datenbank abrufen.
`-new`	Neuen Datensatz hinzufügen.
`-scriptnames`	Namen aller verfügbaren Scripts für eine bereitgestellte und im Web gemeinsam genutzte Datenbank abrufen.
`-view`	Layout-Informationen aus einer Datenbank abrufen, wenn die `FMPXMLLAYOUT`-Grammatik angegeben ist. Ruft den Abschnitt `<metadata>` des XML-Dokuments und eine leere Datensatzmenge ab, wenn die `fmresultset`- oder `FMPXMLRESULT`-Grammatik angegeben ist.

Verwenden Sie diese Query-Parameternamen	Mit diesen Query-Befehlen
`-db (Datenbankname)`	Erforderlich für alle Query-Befehle mit Ausnahme von `-dbnames`
`-delete.related`	Optional bei `-edit`
`-field`	Erforderlich zur Angabe eines Felds in einem URL für Containerabfragen. Weitere Informationen finden Sie unter „Erläuterung der URL-Syntax für FileMaker-Containerobjekte in XML-Lösungen“ auf Seite 25.
`fieldname`	Mindestens ein Feldname ist für `-edit` erforderlich. Optional bei `-find`. Weitere Informationen finden Sie unter „Query-Parameter „fieldname“ (Nicht-Containerfeldname)“ auf Seite 56.
`fieldname.op (Operator)`	Optional bei `-find`
`-lay (Layoutname)`	Erforderlich für alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.
`-lay.response` (Layout für XML-Antwort wechseln)	Optional für alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.
`-lop` (logischer Operator)	Optional bei `-find`
`-max` (maximale Datensätze)	Optional bei `-find`, `-findall` und `-findquery`
`-modid` (Änderungs-ID)	Optional bei `-edit`
`-query`	Erforderlich bei zusammengesetzten `-findquery`-Suchabfragen
`-recid` (Datensatz-ID)	Erforderlich bei `-edit`, `-delete`, `-dup`. Optional bei `-find`
`-relatedsets.filter`	Optional bei `-find`, `-findall`, `-findany`, `-edit`, `-new`, `-dup` und `-findquery`
`-relatedsets.max`	Optional bei `-find`, `-edit`, `-new`, `-dup` und `-findquery`
`-script` (Script ausführen)	Optional bei `-find`, `-findall`, `-findany`, `-new`, `-edit`, `-delete`, `-dup`, `-view` und `-findquery`
`-script.param` (einen Parameterwert an das durch `-script` angegebene Script übergeben)	Optional bei `-script` und `-findquery`

Verwenden Sie diese Query-Parameternamen	Mit diesen Query-Befehlen
`-script.prefind` (Script vor `-find`, `-findany` und `-findall` ausführen)	Optional bei `-find`, `-findany`, `-findall` und `-findquery`
`-script.prefind.param` (einen Parameterwert an das durch `-script.prefind` angegebene Script übergeben)	Optional bei `-script.prefind` und `-findquery`
`-script.presort` (Script vor Sortierung ausführen)	Optional bei `-find`, `-findall` und `-findquery`
`-script.presort.param` (einen Parameterwert an das durch `-script.presort` angegebene Script übergeben)	Optional bei `-script.presort` und `-findquery`
`-skip` (Datensätze überspringen)	Optional bei `-find`, `-findall` und `-findquery`
`-sortfield.[1-9]` (Sortierfeld)	Optional bei `-find`, `-findall` und `-findquery`
`-sortorder.[1-9]` (Sortierfolge)	Optional bei `-find`, `-findall`

Wechseln des Layouts für eine XML-Antwort

Der Query-Parameter `-lay` gibt das Layout an, das Sie beim Abfragen von XML-Daten verwenden wollen. Häufig eignet sich dasselbe Layout für die Verarbeitung der Daten, die von der Abfrage geliefert werden. In einigen Fällen sollten Sie die Daten in einem Layout durchsuchen, das Felder enthält, die aus Sicherheitsgründen in keinem Layout vorhanden sind, mit dem die Ergebnisse angezeigt werden. (Um Daten in einem Feld zu suchen, muss das Feld auf das Layout gesetzt werden, das Sie in der XML-Abfrage angeben.)

Um für die Anzeige der XML-Antwort ein anderes Layout als für die Verarbeitung der XML-Abfrage zu verwenden, können Sie den optionalen Query-Parameter `-lay.response` angeben.

Die folgende Abfrage sucht z. B. Werte größer 100.000 im Feld „Salary“ im Layout „Budget“. Die gefundenen Daten werden im Layout „ExecList“ angezeigt, das das Feld „Salary“ nicht enthält.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=Budget&Salary=100000&Salary.op=gt&-find&-lay.response=ExecList
```

Erläuterung der Verarbeitung einer XML-Abfrage

Es gibt mehrere Query-Parameter, die die Verarbeitung einer XML-Abfrage und die Generierung eines XML-Dokuments beeinflussen.

FileMaker Server und die Web Publishing Engine verarbeiten eine XML-Abfrage in dieser Reihenfolge:

1. Verarbeitung des `-lay` -Query-Parameters
2. Festlegen der globalen Feldwerte, die in der Query angegeben sind (der Teil `.global=` des URLs)
3. Verarbeitung des Query-Parameters `-script.prefind`, falls angegeben
4. Verarbeitung der Query-Befehle wie z. B. `-find` oder `-new`
5. Verarbeitung des Query-Parameters `-script.presort`, falls angegeben

6. Sortieren der Ergebnisdaten, falls eine Sortierung angegeben wurde
7. Verarbeitung des Query-Parameters `-script`, falls angegeben
8. Verarbeitung des Query-Parameters `-lay.response`, um das Layout zu wechseln, falls angegeben
9. Generieren des XML-Dokuments.

Wenn einer der obigen Schritte einen Fehlercode erzeugt, stoppt die Abfrageverarbeitung und etwaige nachfolgende Schritte werden nicht ausgeführt. Jedoch bleiben alle vorangehenden Schritte in der Abfrage ausgeführt.

Betrachten Sie z. B. eine Abfrage, die den aktuellen Datensatz löscht, die Datensätze sortiert und dann ein Script ausführt. Wenn der Parameter `-sortfield` ein nicht vorhandenes Feld angibt, löscht die Abfrage den aktuellen Datensatz und gibt dann Fehlercode 102 („Feld fehlt“) zurück, führt aber das Script nicht aus.

Fehlerbehebung beim XML-Dokumentzugriff

Wenn beim Zugriff auf XML-Dokumente mit der Web Publishing Engine Probleme auftreten, prüfen Sie Folgendes:

- Die erweiterten Zugriffsrechte in der Datenbank müssen auf „XML Custom Web Publishing“ eingestellt und einem Benutzerkonto zugewiesen sein. Weitere Informationen finden Sie unter „Aktivieren von Instant Web Publishing in einer Datenbank“ auf Seite 14.
- Die Datenbank wird auf der Datenbank-Server-Komponente des FileMaker Server-Einsatzes bereitgestellt und von FileMaker Server geöffnet. Weitere Informationen finden Sie in der FileMaker Server Hilfe.
- Prüfen Sie, ob der verwendete Datenbankkontoname und ggf. Ihr Passwort korrekt sind.
- Die Web-Server-Komponente des FileMaker Server-Einsatzes wird ausgeführt.
- Die Web Publishing Engine-Komponente des FileMaker Server-Einsatzes wird ausgeführt.
- XML Publishing ist in der Web Publishing Engine-Komponente aktiviert. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

Kapitel 5
Einstellen, Testen und Überwachen einer Site

Dieses Kapitel enthält Informationen über das Einstellen und Testen einer Custom Web Publishing-Site vor ihrem Einsatz in einer Produktionsumgebung. Zudem finden Sie hier Informationen zur Verwendung von Protokolldateien, um die Site während der Tests und nach der Freigabe zu überwachen.

Einstellen einer Custom Web Publishing-Site

Bevor Sie Ihre Site richtig testen können, müssen Sie die erforderlichen Dateien an die richtigen Speicherorte auf dem bzw. den Servern kopieren oder verschieben.

So stellen Sie Ihre Site zum Testen ein und bereiten Sie dafür vor:

1. Folgen Sie den Anleitungen in Kapitel 3, „Vorbereiten von Datenbanken für Custom Web Publishing“.
2. Stellen Sie sicher, dass XML in FileMaker Server Admin Console aktiviert und richtig konfiguriert wurde.

   Hinweis Weitere Informationen finden Sie in der FileMaker Server Hilfe.
3. Stellen Sie sicher, dass der Web-Server und die Web Publishing Engine laufen.
4. Kopieren oder verschieben Sie referenzierte Containerobjekte auf den Web-Server-Rechner.
   - Falls die Containerfelder die tatsächlichen Dateien in der FileMaker-Datenbank speichern, ist mit dem Inhalt der Containerfelder keine Aktion erforderlich, wenn die Datenbank richtig auf der Datenbank-Server-Komponente des FileMaker Server-Einsatzes bereitgestellt und zugänglich ist.
   - Wenn ein Datenbank-Containerfeld einen Dateiverweis anstelle der eigentlichen Datei speichert, muss das referenzierte Containerobjekt im FileMaker Pro-Web-Ordner gespeichert sein, wenn der Datensatz erstellt oder bearbeitet wird. Um Ihre Site einzustellen, kopieren oder verschieben Sie die referenzierten Container in einen Ordner mit dem gleichen relativen Speicherort im Root-Ordner der Web-Server-Software.
   - Wenn ein Datenbank-Containerfeld das Containerobjekt extern speichert, verwenden Sie den Assistenten für den Datenbank-Upload, um die Datenbankdatei und Containerfeld-Objekte vom Dateisystem Ihres Computers zu FileMaker Server zu übertragen. Wenn Sie eine Datenbank, die ein Containerfeld mit extern gespeicherten Objekten verwendet, manuell hochladen, müssen Sie die referenzierten Objekte in einen Unterordner des Ordners RC_Data_FMS, wie in „Containerfelder mit extern gespeicherten Daten“ auf Seite 17 beschrieben, kopieren.
5. Kopieren Sie etwaige zusätzliche Komponenten Ihrer Web-Anwendung auf den Web-Server-Rechner. Ihre Web-Anwendung verarbeitet die XML-Daten, bevor sie sie an eine andere Anwendung oder den Client sendet.

Testen einer Custom Web Publishing-Site

Bevor Sie Benutzern mitteilen, dass Ihre Custom Web Publishing-Site verfügbar ist, sollten Sie überprüfen, ob Aussehen und Funktionsweise Ihren Vorstellungen entsprechen.

- Testen Sie Funktionen wie das Suchen, Hinzufügen, Löschen und Sortieren von Datensätzen mit unterschiedlichen Konten und Zugriffsrechten.
- Überprüfen Sie, ob sich die Berechtigungen wie erwartet verhalten, indem Sie sich mit verschiedenen Konten anmelden. Stellen Sie sicher, dass nicht autorisierte Benutzer nicht auf Ihre Daten zugreifen oder sie ändern können.
- Überprüfen Sie alle Scripts, um sicherzustellen, dass das Ergebnis wie erwartet ist (siehe „FileMaker-Scripts und Custom Web Publishing“ auf Seite 18 für Informationen zur Gestaltung Web-freundlicher Scripts).
- Testen Sie Ihre Site mit unterschiedlichen Betriebssystemen und Webbrowsern.

Hinweis Wenn Sie über keine Netzwerkverbindung verfügen und FileMaker Server – also Web-Server, Web Publishing Engine und Datenbank-Server – auf einem einzigen Computer installiert haben, können Sie Ihre Custom Web Publishing-Site testen, indem Sie im URL Folgendes verwenden: http://127.0.0.1/. Informationen zur URL-Syntax finden Sie unter „Erläuterung der URL-Syntax für XML-Daten und Containerobjekte“ auf Seite 24

Beispiele für Stylesheets zum Testen der XML-Ausgabe

Hier sind zwei Beispiele für XSLT-Stylesheets, die sich zum Testen von XML-Ausgabe eignen.

- Das folgende Stylesheet-Beispiel gibt die angeforderten XML-Daten ohne Umwandlung aus. Dieses Stylesheet ist nützlich für die Anzeige der tatsächlichen XML-Daten, die die Web Publishing Engine verwendet.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
    <xsl:output method="xml"/>
    <xsl:template match="/">
        <xsl:copy-of select="."/>
    </xsl:template>
</xsl:stylesheet>
```

- Bei der Fehlerbehebung in einem Stylesheet können Sie das folgende Beispiel des HTML-Tags <textarea> verwenden, um das XML-Quelldokument anzuzeigen, auf das über das Stylesheet in einem abrollbaren Textbereich zugegriffen wurde.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:fmrs="http://www.filemaker.com/xml/fmresultset">
    <xsl:output method="html"/>
<html>
    <Datenbereich>
    <xsl:template match="/fmrs:fmresultset">
        <textarea rows="20" cols="100">
            <xsl:copy-of select="."/>
        </textarea><br/>
    </xsl:template>
    </body>
</html>
</xsl:stylesheet>
```

Überwachen Ihrer Website

Sie können anhand der folgenden Typen von Protokolldateien Ihre Custom Web Publishing-Site überwachen und Informationen über Web-Benutzer sammeln, die Ihre Site besuchen:

- Web-Server-Zugriff und Fehlerprotokolle
- Web Publishing Engine-Protokoll
- Web-Server-Modul-Fehlerprotokoll
- Tomcat-Protokolle

Verwenden der Web-Server-Zugriffs- und Fehlerprotokolle

IIS (Windows): Der Microsoft IIS-Web-Server generiert eine Zugriffsprotokolldatei und zeigt Fehler in der Windows-Ereignisanzeige, anstatt sie in eine Protokolldatei zu schreiben. Die Zugriffsprotokolldatei, standardmäßig im W3C Extended Log File Format, ist eine Aufzeichnung aller beim Web-Server eingehenden HTTP-Abfragen. Sie können auch das W3C Common Logfile Format für das Zugriffsprotokoll verwenden. Weitere Informationen finden Sie in der Dokumentation zum Microsoft IIS-Web-Server.

Apache (Mac OS): Der Apache-Web-Server generiert eine Zugriffsprotokolldatei und eine Fehlerprotokolldatei. Die Apache-Zugriffsprotokolldatei, standardmäßig im W3C Common Logfile Format, ist eine Aufzeichnung aller beim Web-Server eingehenden HTTP-Abfragen. Das Apache-Fehlerprotokoll ist eine Aufzeichnung von Problemen bei der Verarbeitung von HTTP-Abfragen. Weitere Informationen über diese Protokolldateien finden Sie in der Dokumentation zu Ihrem Apache-Web-Server.

Informationen über das W3C Common Logfile Format und das W3C Extended Log File Format finden Sie auf der Website des World Wide Web Consortium: http://www.w3.org.

Verwenden des Web Publishing Engine-Protokolls

Standardmäßig generiert die Web Publishing Engine eine Protokolldatei namens wpe.log, die Einträge zu aufgetretenen Web Publishing Engine-Fehlern einschließlich Anwendungs-, Nutzungs- und Systemfehlern enthält. Sie können die Web Publishing Engine zudem Informationen in Bezug auf Custom Web Publishing wie Endbenutzer-XML-Abfragen einschließen lassen, um Web-Publishing-Ausgabe oder Änderungen an den Custom Web Publishing-Einstellungen generieren zu lassen.

Die Datei wpe.log befindet sich im folgenden Verzeichnis in der Web Publishing Engine-Komponente Ihres FileMaker Server-Einsatzes:

- **IIS (Windows):**
  `<Laufwerk>:\Programme\FileMaker\FileMaker Server\Logs\wpe.log`
  Dabei ist `<Laufwerk>` das Hauptlaufwerk, von dem Ihr System startet.
- **Apache (Mac OS):** `/Library/FileMaker Server/Logs/wpe.log`

Web Publishing Engine-Protokolleinstellungen

Die Datei wpe.log wird generiert, wenn die Option **Protokollierung für Custom Web Publishing aktivieren** in der Admin Console aktiviert ist.

Protokollierungsoption aktiviert	In wpe.log protokollierte Informationen
Fehlerstufemeldungen	Alle aufgetretenen Web Publishing Engine-Fehler einschließlich Anwendungs-, Nutzungs- und Systemfehlern.
Info- und Fehlerstufemeldungen	Alle oben beschriebenen Fehler und Informationen zum Zugriff auf die Web Publishing Engine. Es enthält einen Eintrag aller Endbenutzer-XML-Abfragen, um Custom Web Publishing-Ausgabe zu generieren.

Die Einstellung **Fehlerstufemeldungen** ist standardmäßig aktiviert. Informationen zur Einstellung dieser Optionen über die Admin Console finden Sie in der FileMakerServer Hilfe.

Hinweis Für Custom Web Publishing mit FileMaker Server 12 ersetzt die Datei wpe.log file die Dateien wpc_access_log.txt und pe_application_log.txt früherer Versionen.

Wichtig Mit der Zeit wird die Datei wpe.log ggf. sehr groß. Verwenden Sie die Admin Console, um die maximale Größe der Datei wpe.log festzulegen. Wenn die Datei wpe.log diese maximale Größe erreicht, kopiert die Web Publishing Engine die Datei wpe.log in eine einzelne Sicherungsdatei wpe.log.1 und erstellt eine neue Datei wpe.log. Speichern Sie eine Archivkopie der Datei wpe.log.1 regelmäßig, wenn Sie mehr als eine Sicherungskopie wünschen.

Web Publishing Engine-Protokollformat

Die Datei wpe.log file verwendet das folgende Format für die Einträge:

```
[ZEITSTEMPEL_GMT] [WPC_HOSTNAME] [CLIENT_IP:PORT] [KONTONAME] [MODULTYP]
[SCHWERE] [FM_FEHLERCODE] [RÜCKGABE_BYTE] [MELDUNG]
```

Dabei gilt:

- `[ZEITSTEMPEL_GMT]` sind Datum und Zeit des Eintrags in Greenwich Mean Time (GMT).
- `[WPC_HOSTNAME]` ist der Name des Rechners, auf dem die Web Publishing Engine installiert ist.
- `[CLIENT_IP:PORT]` ist die IP-Adresse und der Port des Clients, von dem die XML-Abfrage ausging.

- [KONTONAME] ist der Kontoname, der für die Protokollierung in der bereitgestellten FileMaker-Datenbank verwendet wird.
- [MODULTYP] ist entweder: XML für Custom Web Publishing mit XML-Abfragen oder PHP für Custom Web Publishing mit PHP-Abfragen.
- [SCHWERE] ist entweder INFO, das eine informative Meldung angibt, oder ERROR, das eine Fehlermeldung angibt.
- [FM_FEHLERCODE] ist die für eine Fehlermeldung zurückgegebene Fehlernummer. Die Fehlernummer kann ein Fehlercode für FileMaker-Datenbanken sein (siehe „Fehlercodenummern für FileMaker-Datenbanken“ auf Seite 67).

  Zusätzlich kann die Fehlernummer eine HTTP-Fehlernummer sein, der die „HTTP:“-Zeichenfolge vorangestellt ist.
- [RÜCKGABE_BYTE] ist die Anzahl an Bytes, die die Anfrage zurückgibt.
- [MELDUNG] gibt zusätzliche Informationen zu dem Protokolleintrag.

Beispiele für Web Publishing Engine-Protokollmeldungen

Die folgenden Beispiele zeigen die Arten der Meldungen, die in einer wpe.log-Datei auftreten können:

- Wenn die Web Publishing Engine startet oder anhält

```
2012-06-02 15:15:31 -0700 - - - - INFO - - FileMaker Server Web Publishing Engine started.
2012-06-02 15:46:52 -0700 - - - - INFO - - FileMaker Server Web Publishing Engine stopped.
```

- Erfolgreiche oder fehlgeschlagene XML-Query-Abfragen

```
2012-06-02 15:21:08 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML INFO 0 3964 "/fmi/xml/fmresultset.xml?-db=Contacts&-lay=Contact_Details&-findall"
2012-06-02 15:26:31 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML ERROR 5 596 "/fmi/xml/fmresultset.xml?-db=Contacts&-layout=Contact_Details&-findall"
```

- Scriptfehler

```
2012-06-02 17:33:12 -0700 WPC_SERVER 192.168.100.101:0 jdoe - ERROR 4 - Web Scripting Error: 4, File: "10b_MeetingsUpload", Script: "OnOpen", Script Step: "Show Custom Dialog"
```

- Änderungen an den Custom Web Publishing-Einstellungen

```
2012-06-09 10:59:49 -0700 WPC_SERVER 192.168.100.101:0 jdoe - INFO - - XML Web Publishing Engine is enabled.
```

- Systemfehler

```
2012-06-02 15:30:42 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML ERROR - - Communication failed
```

Verwenden des Web-Server-Modul-Fehlerprotokolls

Wenn der Web-Server keine Verbindung zur Web Publishing Engine aufbauen kann, generiert das Web Server Modul eine Protokolldatei, die alle Fehler in Zusammenhang mit ihrer Ausführung aufzeichnet. Diese Datei heißt web_server_module_log.txt und befindet sich im Ordner „Logs“ im FileMaker Server-Ordner auf dem Web-Server-Host.

Verwenden der Tomcat-Protokolle

Wenn FileMaker Server ein von einem internen Web-Server-Fehler verursachtes Problem feststellt, finden Sie die Tomcat-Protokolle evtl. hilfreich. Die Tomcat-Protokolle befinden sich in der Web-Server-Komponente des FileMaker Server-Einsatzes:

- **IIS (Windows):** `<Laufwerk>:\Programme\FileMaker\FileMaker Server\Admin\admin-master-tomcat\logs\`
  Dabei ist `<Laufwerk>` das Hauptlaufwerk, von dem Ihr System startet.
- **Apache (Mac OS):** `/Library/FileMaker Server/Admin/admin-master-tomcat/logs/`

Anhang A
Gültige Namen in Query-Strings

Dieser Anhang beschreibt die gültigen Namen für Query-Befehle und deren Parameter, die Sie in einem Query-String verwenden können, wenn Sie über die Web Publishing Engine auf FileMaker-Daten zugreifen.

Erläuterung von Query-Befehlen und -Parametern

Die folgende Liste enthält sämtliche Namen für Query-Befehle und Query-Parameter:

Query-Befehlsnamen	Query-Parameternamen
`-dbnames` (siehe Seite 52.)	`-db` (siehe Seite 55.)
`-delete` (siehe Seite 52.)	`-field` (siehe Seite 56.)
`-dup` (siehe Seite 52.)	`fieldname` (siehe Seite 56.)
`-edit` (siehe Seite 52.)	`fieldname.op` (siehe Seite 57.)
`-find`, `-findall`, `-findany` (siehe Seite 53.)	`-lay` (siehe Seite 58.)
`-findquery` (siehe Seite 53.)	`-lay.response` (siehe Seite 58.)
`-layoutnames` (siehe Seite 54.)	`-lop` (siehe Seite 58.)
`-new` (siehe Seite 54.)	`-max` (siehe Seite 58.)
`-scriptnames` (siehe Seite 54.)	`-modid` (siehe Seite 59.)
`-view` (siehe Seite 55.)	`-query` (siehe Seite 59.)
	`-recid` (siehe Seite 60.)
	`-relatedsets.filter` (siehe Seite 61.)
	`-relatedsets.max` (siehe Seite 62.)
	`-script` (siehe Seite 62.)
	`-script.param` (siehe Seite 62.)
	`-script.prefind` (siehe Seite 63.)
	`-script.prefind.param` (siehe Seite 63.)
	`-script.presort` (siehe Seite 63.)
	`-script.presort.param` (siehe Seite 64.)
	`-skip` (siehe Seite 64.)
	`-sortfield.[1-9]` (siehe Seite 64.)
	`-sortorder.[1-9]` (siehe Seite 65.)

Wichtig Der Parameter `-lay` für die Angabe eines Datenbanklayouts ist bei allen Query-Befehlen erforderlich mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Richtlinien für die Verwendung von Query-Befehlen und -Parametern

Beachten Sie folgende Richtlinien, wenn Sie Query-Befehle und -Parameter in einem Query-String verwenden:

- Ein Query-String darf nur einen einzigen Query-Befehl enthalten. Ein Query-String kann z. B. `-new` enthalten, um einen neuen Datensatz hinzuzufügen, aber derselbe Query-String kann nicht gleichzeitig `-new` und `-edit` enthalten.

- Die meisten Query-Befehle verlangen mehrere entsprechende Query-Parameter im Query-String. Beispielsweise verlangen alle Query-Befehle außer `-dbnames` den Parameter `-db`, der die abzufragende Datenbank angibt. (Siehe dazu die Tabellen der erforderlichen Parameter in „Abfragen von XML-Daten mithilfe von FileMaker-Query-Strings“ auf Seite 37)
- Geben Sie für Query-Parameter und Feldnamen den jeweils gewünschten Wert an, z. B. `-db=employees`. Geben Sie für Query-Befehle kein `=`-Zeichen oder einen Wert nach dem Befehlsnamen wie `-findall` an.
- Die Web Publishing Engine konvertiert alle reservierten Wörter in Kleinbuchstaben, hierzu zählen Query-Befehle, Query-Parameter und Befehlswerte, bei denen spezifische Werte erwartet werden (zum Beispiel: `-lop=and`, `-lop=or`, `-sortorder=ascend`, `-sortorder=descend`, `-max=all`).
- Datenbanknamen, Layoutnamen und Feldnamen, die in Query-Strings verwendet werden, unterscheiden keine Groß- und Kleinschreibung, z. B. gibt `-lay=meinlayout` den Layoutnamen `MeinLayout` an.
- Es empfiehlt sich nicht, Punkte oder Klammern in Feldnamen zu verwenden. In bestimmten Fällen funktionieren Feldnamen mit Punkten, aber Feldnamen mit den folgenden Ausnahmen können nie verwendet werden:
  - Dem Punkt darf keine Zahl folgen. Beispielsweise ist `meinfeld.9` ein ungültiger Dateiname.
  - Dem Punkt darf nicht die Textzeichenfolge `op` (die beiden Buchstaben „o“ und „p“) folgen. Beispielsweise ist `meinfeld.op` ein ungültiger Feldname.
  - Dem Punkt darf nicht die Textzeichenfolge `global` (das Wort „global“) folgen. Beispielsweise ist `meinfeld.global` ein ungültiger Feldname.

  Auf Feldnamen, die eine dieser Ausnahmen enthalten, kann nicht über XML mithilfe einer HTTP-Query zugegriffen werden. Diese Konstrukte sind für Datensatz-IDs wie in Abschnitt „Erläuterung der Syntax für einen voll qualifizierten Feldnamen“ nachfolgend beschrieben reserviert.
- Für den Befehl `-find` wird die Groß-/Kleinschreibung eines Feldwerts nicht berücksichtigt. Sie können beispielsweise `Field1=Blue` oder `Field1=blue` verwenden. Für die Befehle `-new` und `-edit` wird die verwendete Schreibung für einen Feldwert beibehalten und exakt so in der Datenbank gespeichert wie im Query-String angegeben. Beispiel: `LastName=Doe`.

Query-Befehls-Parsing

Die Web Publishing Engine analysiert Query-Befehle in der folgenden Reihenfolge und stoppt die Analyse von XML-Queries mit dem ersten Fehler. Wenn ein Fehlercode zurückgegeben wird, entspricht der Fehlercode dem ersten Fehler, der erkannt wird.

1. Hat die Query einen Befehl und ist der Query-Befehl gültig?

Es ist ein Fehler, wenn der Query der Befehl fehlt oder sie einen unbekannten Befehl verwendet. Beispiel: `-database`

2. Hat die Query zwei Befehle?

Beispiel: `-find&-edit`

3. Hat die Query einen ungültigen Wert für einen Befehl oder Parameter?

Beispiel: `-lop=amd`

4. Fehlt der Query der erforderliche Datenbanknamen-Parameter (`-db`)?

5. Fehlt der Query der erforderliche Layoutbanknamen-Parameter (`-lay`)?

6. Hat die Query eine ungültige Sortierung?

7. Hat die Query ungültige Feldparameter?

Hinweis Wenn eine Query gültige, aber irrelevante Informationen enthält, wird die Query ohne Fehler verarbeitet. Wenn Sie z. B. den Parameter `-lop` bei einem `-delete`-Befehl angeben, wird der Parameter `-lop` ignoriert, da er nicht dazu führt, dass die Query ungültig oder zweideutig wird.

Informationen zu spezifischen zurückgegebenen Fehlercodes finden Sie unter Anhang B, „Fehlercodes für Custom Web Publishing“.

Erläuterung der Syntax für einen voll qualifizierten Feldnamen

Ein vollständig qualifizierter Feldname identifiziert eine exakte Instanz eines Felds. Da Felder mit gleichen Namen auf unterschiedlichen Tabellen basieren können, müssen Sie vollständig qualifizierte Namen verwenden, um Fehler zu vermeiden.

Syntax für einen vollständig qualifizierten Feldnamen:

```
tabellen-name::feld-name(wiederholung-nummer).datensatz-id
```

Dabei gilt:

- `tabellen-name` ist der Name der Tabelle, die das Feld enthält. Der Tabellenname ist nur erforderlich, wenn sich das Feld nicht in der zugrunde liegenden Tabelle des im Query-String angegebenen Layouts befindet.
- `feld-name(wiederholung-nummer)` ist der spezifische Wert in einem Wiederholfeld und ist nur für Wiederholfelder erforderlich. Die Wiederholungsnummer wird ab der Ziffer 1 gezählt. Beispielsweise gibt `feld-name(2)` den zweiten Wert in einem Wiederholfeld an. Wenn Sie keine Wiederholungsnummer für ein Wiederholfeld angeben, wird der erste Wert des Wiederholfelds verwendet. Die Wiederholungsnummer ist für die Query-Befehle `-new` und `-edit` erforderlich, wenn sie Wiederholfelder verwenden, jedoch nicht für den Befehl `-find`.
- `datensatz-id` gibt die Datensatz-ID an. Diese ist nur erforderlich, wenn Sie einen Query-String verwenden, um Datensätze in einem Ausschnitt hinzuzufügen oder zu bearbeiten. Weitere Informationen finden Sie in den Abschnitten „Hinzufügen von Datensätzen in einem Ausschnitt“ und „Bearbeiten von Datensätzen in einem Ausschnitt“. Die `record-id` ist für die Query-Befehle `-new` und `-edit` erforderlich, wenn sie Ausschnittfelder verwenden, jedoch nicht für den Befehl `-find`.

Hinweis Damit sie verfügbar sind, müssen Felder auf das Layout gesetzt werden, das Sie im Query-String angeben.

Verwenden von Query-Befehlen für Ausschnittfelder

Die folgenden Abschnitte beschreiben, wie Query-Befehle für Ausschnittfelder funktionieren.

Hinzufügen von Datensätzen in einem Ausschnitt

Um einem Ausschnitt gleichzeitig mit dem Hinzufügen eines Ausgangsdatensatzes einen neuen Datensatz hinzuzufügen, verwenden Sie den Query-Befehl `-new` und führen den folgenden Query-String für die Abfrage aus:

- Verwenden Sie den vollständig qualifizierten Feldnamen für das Bezugsfeld im Ausschnitt.
- Geben Sie 0 als Datensatz-ID nach dem Namen des Bezugsfelds im Ausschnitt an.
- Geben Sie mindestens eines der Felder für den Ausgangsdatensatz an, bevor Sie das Bezugsfeld im Ausschnitt angeben.
- Geben Sie die Daten für das Abgleichsfeld (Schlüsselfeld) im Ausgangsdatensatz an.

Beispiel: Der folgende URL fügt einen neuen Employee-Ausgangsdatensatz für John Doe und gleichzeitig einen neuen Bezugsdatensatz für Jane im Ausschnitt hinzu. Der Name der Bezugstabelle ist „Dependents“ und der Name des Bezugsfelds im Ausschnitt lautet „Names“. Das Abgleichsfeld ID speichert eine Angestelltenkennung.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=family&FirstName=John&LastName=Doe&ID=9756&Dependents::Names.0=Jane&-new
```

Hinweis Sie können pro Abfrage nur einen Bezugsdatensatz hinzufügen.

Bearbeiten von Datensätzen in einem Ausschnitt

Zur Bearbeitung eines oder mehrerer Datensätze in einem Ausschnitt verwenden Sie den Befehl `-edit` und eine Datensatz-ID, um den Ausgangsdatensatz anzugeben, der die zu bearbeitenden Ausschnitt-Datensätze enthält. Geben Sie den entsprechenden zu bearbeitenden Ausschnitt-Datensatz anhand seiner Datensatz-ID in einem vollständig qualifizierten Feldnamen an. Sie können eine Datensatz-ID dem Datensatz-ID-Attribut des `<record>`-Elements im `<relatedset>`-Element in den XML-Daten entnehmen. Weitere Informationen finden Sie unter „Verwenden der fmresultset-Grammatik“ auf Seite 29

Beispiel: Der folgende URL bearbeitet einen Datensatz in einem Ausschnitt, wobei der Ausgangsdatensatz die Datensatz-ID 1001 hat. Der Name der Bezugstabelle ist „Dependents“, der Name des Bezugsfelds im Ausschnitt lautet „Names“ und die 2 in „Names.2“ ist die Datensatz-ID des Ausschnitt-Datensatzes.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=1001&Dependents::Names.2=Kevin&-edit
```

Das folgende Beispiel zeigt, wie Sie eine Abfrage verwenden können, um mehrere Ausschnitt-Datensätze über den Ausgangsdatensatz zu bearbeiten:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=1001&Dependents::Names.2=Kevin&Dependents::Names.5=Susan&-edit
```

Sie können auch den Befehl `-edit` verwenden und 0 als ID des Ausschnitt-Datensatzes angeben, um dem Ausschnitt einen neuen Bezugsdatensatz für einen bestehenden Ausgangsdatensatz hinzuzufügen. Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=1001&Dependents::Names.0=Timothy&-edit
```

Löschen von Ausschnittdatensätzen

Um Ausschnittdatensätze zu löschen, verwenden Sie den Parameter `-delete.related` mit dem Befehl `-edit` anstelle des Befehls `-delete`.

Zum Beispiel löscht der folgende URL den Datensatz „1001“ aus der Tabelle „employees“:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=1001&-delete
```

Aber der folgende URL löscht einen Ausschnittdatensatz mit einer Datensatz-ID „3“ aus der Bezugstabelle „Dependents“ mit der Ausgangsdatensatz-ID „1001“.

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=1001&-delete.related=Dependents.3&-edit
```

Weitere Informationen finden Sie unter „–delete.related (Löschen von Ausschnittdatensätzen, Query-Parameter)“ auf Seite 55

Abfragen von Ausschnittfeldern

In einer Lösung mit vielen Datensätzen kann das Abfragen und Sortieren von Datensätzen einige Zeit in Anspruch nehmen. Um die anzuzeigende Anzahl von Datensätzen und Zeilen in einer Bezugsmenge einzuschränken, verwenden Sie die Parameter `-relatedsets.filter` und `-relatedsets.max` bei Suchabfragen. Weitere Informationen finden Sie unter „–relatedsets.filter (Ausschnittdatensätze filtern, Query-Parameter)“ auf Seite 61 und „–relatedsets.max (Ausschnittdatensätze beschränken, Query-Parameter)“ auf Seite 62

Erläuterung der Syntax für das Angeben eines Variablenfelds

Syntax für das Angeben eines Variablenfelds:

```
tabellen-name::feld-name(wiederholung-nummer).global
```

Dabei gilt: `global` gibt ein Feld an, das global gespeichert wird. Informationen über `tabellen-name` und `feld-name(wiederholung-nummer)` finden Sie unter „Erläuterung der Syntax für einen voll qualifizierten Feldnamen“ auf Seite 49 Informationen über Variablenfelder finden Sie in der FileMaker Pro Hilfe.

Sie müssen die `.global`-Syntax verwenden, um ein Variablenfeld in einem Query-String zu identifizieren. Die Web Publishing Engine setzt die Parameterwerte für Variablenfelder, bevor sie den Query-Befehl ausführt oder andere Parameterwerte im Query-String setzt. Bei direkten XML-Abfragen werden die globalen Werte direkt nach Ausführung der Abfrage ungültig.

Wenn Sie ein Variablenfeld in einem Query-String nicht mit der `.global`-Syntax identifizieren, berechnet die Web Publishing Engine das Variablenfeld zusammen mit dem restlichen Query-String, ohne zuvor den Wert für das Variablenfeld zu setzen.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=departments
&Country.global=USA&-recid=1&-edit
```

Query-Befehlsreferenz

Dieser Abschnitt enthält Informationen über die Query-Befehle, die für XML-Abfragen zur Verfügung stehen.

Query-Befehl -dbnames (Datenbanknamen)

Ruft die Namen aller Datenbanken ab, die FileMaker Server bereitstellt und die für Custom Web Publishing mit XML aktiviert sind.

Erforderliche Query-Parameter: (keine)

Beispiel:

So rufen Sie die Datenbanknamen ab:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-dbnames
```

Query-Befehl -delete (Datensatz löschen)

Löscht den vom `-recid`-Parameter angegebenen Datensatz.

Erforderliche Query-Parameter: `-db`, `-lay`, `-recid`

Optionaler Query-Parameter: `-script`

Beispiel:

So löschen Sie einen Datensatz:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-recid=4&-delete
```

Query-Befehl -dup (Datensatz duplizieren)

Dupliziert den durch `-recid` angegebenen Datensatz.

Erforderliche Query-Parameter: `-db`, `-lay`, `-recid`

Optionaler Query-Parameter: `-script`

Beispiel:

So duplizieren Sie den angegebenen Datensatz:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-recid=14&-dup
```

Query-Befehl -edit (Datensatz bearbeiten)

Aktualisiert den vom Parameter `-recid` angegebenen Datensatz und füllt die Felder mit dem Inhalt etwaiger Feldname-Wert-Paare. Der `-recid`-Parameter gibt an, welcher Datensatz bearbeitet werden sollte.

Erforderliche Query-Parameter: `-db`, `-lay`, `-recid`, ein oder mehrere Feldname(n)

Optionaler Query-Parameter: `-modid`, `-script`, Feldname

Hinweis Informationen über das Bearbeiten von Datensätzen in einem Ausschnitt finden Sie unter „Bearbeiten von Datensätzen in einem Ausschnitt“ auf Seite 50

Beispiel:

So bearbeiten Sie einen Datensatz:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-recid=13&Country=USA&-edit
```

Query-Befehle -find, -findall oder -findany (Datensätze suchen)

Sendet eine Suchabfrage mit definierten Kriterien.

Erforderliche Query-Parameter: `-db`, `-lay`

Optionale Query-Parameter: `-recid`, `-lop`, `-op`, `-max`, `-skip`, `-sortorder`, `-sortfield`, `-script`, `-script.prefind`, `-script.presort`, Feldname

Beispiele:

So suchen Sie einen Datensatz nach Feldname:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=family&Country=USA&-find
```

Hinweis Die mehrfache Angabe eines Feldnamens in einer einzelnen Abfrage wird nicht unterstützt. FileMaker Server geht alle Werte durch, verwendet aber nur den zuletzt geparsten Wert.

So suchen Sie einen Datensatz nach Datensatz-ID:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-recid=427&-find
```

Um alle Datensätze in der Datenbank zu finden, verwenden Sie `-findall`:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-findall
```

Um einen zufällig ermittelten Datensatz zu finden, verwenden Sie `-findany`:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-lay=family
&-findany
```

-findquery (Compound-Suche, Query-Befehl)

Sendet eine Suchabfrage mit mehreren Suchdatensätzen und Datensatzausschluss-Abfragen.

Erforderliche Query-Parameter: `-db`, `-lay`, `-query`

Optionale Query-Parameter: `-max`, `-skip`, `-sortorder`, `-sortfield`, `-script`, `-script.prefind`, `-script.presort`

Beispiel:

Sucht Datensätze mit Hunden oder Katzen, die nicht „Fluffy“ heißen.

```
http://host/fmi/xml/fmresultset.xml?-db=vetclinic&-lay=animals
&-query=(q1);(q2);!(q3)&-q1=typeofanimal&-q1.value=Cat&-q2=typeofanimal
&-q2.value=Dog&-q3=name&-q3.value=Fluffy&-findquery
```

Verwenden des Befehls -findquery für Compound-Suchen

Eine Anweisung `-findquery` besteht aus vier Teilen in der folgenden Reihenfolge:

- Der Parameter `-query`
- Die Query-Abfragedeklarationen, bestehend aus den Query-Bezeichnerdeklarationen und Abfragevorgängen

- Das Suchfeld und Wertedefinitionen für jeden Query-Bezeichner
  - Definieren Sie Query-Bezeichner. Ein Query-Bezeichner ist der Buchstabe „q“ gefolgt von einer Zahl. Zum Beispiel: `-q1`
  - Definieren Sie Query-Bezeichner mit dem Parameter. Beispiel: `-q1.value=fieldvalue`
  - Definieren Sie Query-Bezeichner-Operatoren durch Einschließen als Teil des Ausdrucks `fieldvalue`. So verwenden Sie z. B. einen Stern als Operatir „beginnt mit“: `-q1.value=fieldvalue*`
- Der Befehl `-findquery` am Ende der gesamten Anweisung

Weitere Informationen zur Verwendung des Parameters `-query` finden Sie unter „–query (Compound-Suchabfrage, Query-Parameter)“ auf Seite 59

-layoutnames (Layoutnamen) (Query-Befehl)

Ruft die Namen aller verfügbaren Layouts für eine angegebene Datenbank ab, die FileMaker Server bereitstellt und die für Custom Web Publishing mit XML aktiviert sind.

Erforderliche Query-Parameter: `-db`

Beispiel:

So rufen Sie die Namen verfügbarer Layouts ab:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-layoutnames
```

-new (Neuer Datensatz) (Query-Befehl)

Erzeugt einen neuen Datensatz und fügt in diesen den Inhalt jeglicher Feldnamen/Wert-Paare ein.

Erforderliche Query-Parameter: `-db`, `-lay`

Optionaler Query-Parameter: Ein oder mehrere Feldname(n), `-script`

Hinweis Informationen über das Einfügen neuer Daten für einen Ausschnitt finden Sie unter „Hinzufügen von Datensätzen in einem Ausschnitt“ auf Seite 50

Beispiel:

So fügen Sie einen neuen Datensatz hinzu:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&Country=Australia&-new
```

-scriptnames (Scriptnamen) (Query-Befehl)

Ruft die Namen aller verfügbaren Scripts für eine angegebene Datenbank ab, die FileMaker Server bereitstellt und die für Custom Web Publishing mit XML aktiviert sind.

Erforderliche Query-Parameter: `-db`

Beispiel:

So rufen Sie die Namen aller Scripts ab:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees&-scriptnames
```

Query-Befehl –view (Layoutinformationen anzeigen)

Wenn die FMPXMLLAYOUT-Grammatik angegeben ist: Ruft Layoutinformationen aus einer Datenbank ab und zeigt sie in der FMPXMLLAYOUT-Grammatik an. Wenn eine Datengrammatik (fmresultset oder FMPXMLRESULT) angegeben ist: Ruft den Metadaten-Abschnitt des XML-Dokuments und einen leeren „recordset“ ab.

Erforderliche Query-Parameter: -db, -lay

Optionaler Query-Parameter: -script

Beispiele:

So rufen Sie Layoutinformationen ab:

```
http://192.168.123.101/fmi/xml/FMPXMLLAYOUT.xml?-db=employees
&-lay=departments&-view
```

So rufen Sie Metadaten-Informationen ab:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-view
```

Query-Parameterreferenz

Dieser Abschnitt enthält Informationen über die Query-Parameter, die für XML-Abfragen zur Verfügung stehen.

Query-Parameter –db (Datenbankname)

Gibt die Datenbank an, auf die der Query-Befehl angewendet wird.

Wert: Name der Datenbank ohne etwaige Dateinamenerweiterung.

Hinweis Geben Sie mit dem Namen der Datenbank für den Parameter -db keine Dateinamenerweiterung an. Der tatsächliche Datenbank-Dateiname kann optional eine Erweiterung aufweisen, aber Erweiterungen sind als Wert für den Parameter -db nicht erlaubt.

Erforderlich für: Alle Query-Befehle mit Ausnahme von -dbnames

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-findall
```

-delete.related (Löschen von Ausschnittdatensätzen, Query-Parameter)

Löscht einen Datensatz aus einem Ausschnittfeld.

Optional für: -edit (Query-Befehl)

Erfordert: Ein Bezugstabellenname und eine Datensatz-ID

Beispiel:

Das folgende Beispiel löscht einen Ausschnittdatensatz mit einer Datensatz-ID „20“ aus der Bezugstabelle „jobtable“ mit der Ausgangsdatensatz-ID „7“.

```
http://host/fmi/xml/fmresultset.xml?-db=career&-lay=applications&-recid=7
&-delete.related=jobtable.20&-edit
```

Query-Parameter –field (Containerfeldname)

Gibt den Namen eines Containerfelds zurück.

Erforderlich für: Abfrage von Daten in einem Containerfeld.

Weitere Informationen finden Sie unter „Erläuterung der URL-Syntax für XML-Daten und Containerobjekte“ auf Seite 24

Query-Parameter „fieldname“ (Nicht-Containerfeldname)

Feldnamen dienen als Kontrollkriterien für den Query-Befehl `-find` oder zum Ändern des Inhalts eines Datensatzes. Wenn ein Query-Befehl oder -Parameter einen Wert für ein Nicht-Containerfeld verlangt, verwenden Sie den Feldnamen ohne den Bindestrich (`-`) als den Namensteil des Name-Wert-Paars.

Name: Name des Felds in der FileMaker-Datenbank. Wenn das Feld sich nicht in der zugrunde liegenden Tabelle des Layouts befindet, das im Query-String angegeben ist, muss der Feldname vollständig qualifiziert sein.

Es empfiehlt sich nicht, Punkte oder Klammern in Feldnamen zu verwenden. In bestimmten Fällen funktionieren Feldnamen mit Punkten, aber Feldnamen mit den folgenden Ausnahmen können nie verwendet werden:

- Dem Punkt darf keine Zahl folgen. Beispielsweise ist `meinfeld.9` ein ungültiger Dateiname.
- Dem Punkt darf nicht die Textzeichenfolge `op` (die beiden Buchstaben „o“ und „p“) folgen. Beispielsweise ist `meinfeld.op` ein ungültiger Feldname.
- Dem Punkt darf nicht die Textzeichenfolge `global` (das Wort „global“) folgen. Beispielsweise ist `meinfeld.global` ein ungültiger Feldname.

Auf Feldnamen, die eine dieser Ausnahmen enthalten, kann nicht über XML mithilfe einer HTTP-Query zugegriffen werden. Diese Konstrukte sind für Datensatz-IDs wie in Abschnitt „Erläuterung der Syntax für einen voll qualifizierten Feldnamen“ auf Seite 49 beschrieben reserviert.

Wert: Geben Sie für die Query-Befehle `-new` und `-edit` den Wert an, den Sie im aktuellen Datensatz in dem Feld speichern wollen. Geben Sie für den Query-Befehl `-find` den Wert an, den Sie im Feld suchen wollen. Wenn Sie den Wert für ein Datums-, Zeit- oder Zeitstempelfeld angeben, müssen Sie den Wert gemäß „fm“-Format für den Feldtyp angeben. Die „fm“-Formate sind `MM/tt/jjjj` für Datum, `HH:mm:ss` für Zeit und `MM/tt/jjjj HH:mm:ss` für Zeitstempel.

Erforderlich für: `-edit` (Query-Befehl)

Optional für: `-new`- und `-find`-Query-Befehle

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-op=eq&FirstName=Sam&-max=1&-find
```

Hinweis Die mehrfache Angabe eines Feldnamens in einer einzelnen Abfrage wird nicht unterstützt. FileMaker Server geht alle Werte durch, verwendet aber nur den zuletzt geparsten Wert.

Query-Parameter „fieldname.Op“ (Vergleichsoperator)

Gibt den Vergleichsoperator an, der auf den Feldnamen, der dem Operator vorangeht, angewendet werden soll. Vergleichsoperatoren werden mit dem Query-Befehl `-find` verwendet.

Wert: Der zu verwendende Operator. Der Standard-Operator lautet „begins with“. Gültige Operatoren sind:

Schlüsselwort	Äquivalenter Operator in FileMaker Pro
`eq`	**= Wort**
`cn`	***Wort***
`bw`	**Wort***
`ew`	***Wort**
`gt`	**> Wort**
`gte`	**>= Wort**
`lt`	**< Wort**
`lte`	**<= Wort**
`neq`	**ausschließen, Wort**

Optional für: `-find` (Query-Befehl)

Erfordert: Einen Feldnamen und einen Wert

Syntax für das Angeben eines Vergleichsoperators:

```
tabellen-name::feld-name=wert&tabelle-name::feld-name.op=op-symbol
```

Dabei gilt:

- `tabelle-name` ist die Tabelle, die das Feld enthält, und ist nur erforderlich, wenn das Feld nicht in der Quelltabelle des im Query-String angegebenen Layouts ist.
- `op-symbol` ist eines der Schlüsselwörter in der vorherigen Tabelle, wie z. B. `cn`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&name=Tim&name.op=cn&-find
```

Sie können jeden FileMaker Pro-Suchoperator verwenden, indem Sie das Schlüsselwort `bw` angeben. Um z. B. den Wertebereich über den Bereichsoperator (...) zu ändern, geben Sie das Schlüsselwort `bw` an und setzen dann die Zeichen „...“ vor das Suchkriterium.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&IDnum=915...925&IDnum.op=bw&-find
```

Weitere Informationen über Operatoren für die Suche nach Text finden Sie in der FileMaker Pro Hilfe.

Query-Parameter -lay (Layout)

Gibt das Datenbanklayout an, das Sie verwenden wollen.

Wert: Name des Layouts.

Erforderlich für: Alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-view
```

Query-Parameter -lay.response (Layout wechseln für Antwort)

Gibt an, dass FileMaker Server beim Verarbeiten einer Abfrage das Layout verwenden soll, das der Parameter `-lay` angibt, und bei der Verarbeitung der XML-Antwort in das Layout wechseln soll, das der Parameter `-lay.response` angibt.

Wenn Sie den Parameter `-lay.response` nicht angeben, verwendet FileMaker Server das vom Parameter `-lay` angegebene Layout für die Verarbeitung der Abfrage und der Antwort.

Sie können den Parameter `-lay.response` für XML-Abfragen verwenden.

Wert: Name des Layouts.

Optional für: Alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=Budget&Salary=100000&Salary.op=gt&-find&-lay.response=ExecList
```

Query-Parameter -lop (Logischer Operator)

Gibt an, wie die Suchkriterien im Query-Befehl `-find` als „and“- oder „or“-Suche kombiniert werden.

Wert: `and` oder `or`
Wenn der Query-Parameter `-lop` nicht angegeben wird, verwendet der Query-Befehl `-find` den Wert „and“.

Optional für: `-find` (Query-Befehl)

Hinweis Nicht unterstützt vom Query-Befehl `-findquery`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&Last+Name=Smith&Birthdate=2/5/1972&-lop=and&-find
```

Query-Parameter -max (Maximum Datensätze)

Gibt die maximale Anzahl an Datensätzen an, die zurückgegeben werden soll.

Wert: Eine Zahl oder – für die Rückgabe aller Datensätze – der Wert `all`. Wenn `-max` nicht angegeben wird, werden alle Datensätze zurückgegeben.

Optional für: Query-Befehle `-find`, `-findall` und `-findquery`

Hinweis Der Query-Parameter `-max` wirkt sich nicht auf die für Ausschnittdatensätze zurückgegebenen Werte aus. Um die Anzahl der für Ausschnittdatensätze zurückgegebenen Zeilen einzuschränken, siehe „–relatedsets.max (Ausschnittdatensätze beschränken, Query-Parameter)“ auf Seite 62

Beispiele:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-max=10&-findall
```

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-max=all&-findall
```

Query-Parameter -modid (Änderungs-ID)

Die Änderungs-ID ist ein inkrementeller Zähler, der die aktuelle Version eines Datensatzes angibt. Indem Sie mit dem Query-Befehl `-edit` eine Änderungs-ID angeben, können Sie sicherstellen, dass Sie die aktuelle Version eines Datensatzes bearbeiten. Wenn der angegebene Wert der Änderungs-ID nicht dem aktuellen Wert der Änderungs-ID in der Datenbank entspricht, wird der Query-Befehl `-edit` nicht zugelassen und ein Fehlercode wird zurückgegeben.

Wert: Eine Änderungs-ID, die die aktuelle Version eines Datensatzes in einer FileMaker-Datenbank eindeutig identifiziert.

Optional bei: `-edit` (Query-Befehl)

Erfordert: `-recid`-Parameter

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-recid=22&-modid=6&last_name=Jones&-edit
```

-query (Compound-Suchabfrage, Query-Parameter)

Gibt die Query-Namen und Suchkriterien für eine Compound-Suchabfrage an. Weitere Informationen finden Sie unter „–findquery (Compound-Suche, Query-Befehl)“ auf Seite 53

Wert: Ein Query-Ausdruck.

Erforderlich für: Query-Befehl `-findquery`

Die Syntax für eine Compound-Suchabfrage ist:

```
-query=<request-declarations><request-definitions>&-findquery
```

Wo:

`<request-declarations>` sind zwei oder mehr Abfragedeklarationen.

- Jede Abfragedeklaration setzt sich aus einem oder mehreren Query-Bezeichnern zusammen, die durch Kommata getrennt und in Klammern eingeschlossen sind. Ein Query-Bezeichner ist der Buchstabe „q“ gefolgt von einer Zahl. Beispiel: `q1`
- Eingeschlossen in Klammern fungieren mehrere Queries als logische UND-Suchen, die die Ergebnismenge einengen. Zum Beispiel gibt `(q1, q2)` Datensätze zurück, die `q1` und `q2` entsprechen.

  Hinweis Es empfiehlt sich nicht, die gleichen Felder für mehrere `q`-Variablen in den gleichen „and“-Suchkriterien zu verwenden.

- Wie bei FileMaker Pro kann jede Abfrage eine Such- oder Ausschlussabfrage sein. Eine Suchabfrage fügt die entsprechenden Datensätze der Ergebnismenge hinzu. Eine Ausschlussabfrage entfernt die entsprechenden Datensätze aus der Ergebnismenge. Standard ist eine Suchabfrage. Setzen Sie für eine Ausschlussabfrage ein Ausrufezeichen (!) vor die öffnende Klammer.

  Beispiel: `(q1);!(q2)`

  In diesem Beispiel ist `q1` eine Suchabfrage. `q2` ist eine Ausschlussabfrage, da ihr ein Ausrufezeichen vorangestellt ist.
- Abfragen werden durch Semikolon getrennt. Mehrere Suchabfragen fungieren als logische ODER-Suchen, die die Ergebnismenge erweitern. Zum Beispiel gibt `(q1);(q2)` Datensätze zurück, die `q1` oder `q2` entsprechen. Ausschlussabfragen fungieren nicht als logische ODER-Suchen, da Ausschlussabfragen Datensätze aus der Ergebnismenge entfernen.
- Abfragen werden in der angegebenen Reihenfolge ausgeführt. Die Ergebnismenge umfasst die Ergebnisse aus der gesamten zusammengesetzten Suchabfrage.

`<request-definitions>` ist eine Abfragedefinition für jede Abfragedeklaration. Jede Abfragedefinition besteht aus Suchfeld und Wertdefinition. Ein Minuszeichen (-) beginnt die Abfragedefinition.

Syntax:

```
-<query-id>=<Feldname>&-<query-id>.value=<Wert>
```

Beispiel:

```
-q1=typeofanimal&-q1.value=Cat
-q2=name&-q2.value=Fluffy
```

Beispiel:

Sucht Datensätze mit grauen Katzen, die nicht „Fluffy“ heißen.

```
http://host/fmi/xml/fmresultset.xml?-db=petclinic&-lay=Patients
&-query=(q1, q2);!(q3)&-q1=typeofanimal&-q1.value=Cat&-q2=color
&-q2.value=Gray&-q3=name&-q3.value=Fluffy&-findquery
```

Query-Parameter -recid (Datensatz-ID)

Gibt den Datensatz an, der verarbeitet werden soll. Wird hauptsächlich mit den Query-Befehlen `-edit` und `-delete` verwendet. Vom Befehl `-view` verwendet, um Bezugswertlistendaten in der Grammatik FMPXMLLAYOUT abzurufen.

Wert: Eine Datensatz-ID, die einen Datensatz in einer FileMaker-Datenbank eindeutig angibt.

Erforderlich für: Query-Befehle `-edit`, `-delete` und `-dup`

Optional bei: Befehlen `-find` query und `-view`

Beispiel 1:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-recid=22&-delete
```

Beispiel 2:

```
http://localhost/fmi/xml/FMPXMLLAYOUT.xml?-db=test&-lay=empty&-view&-recid=9
```

-relatedsets.filter (Ausschnittdatensätze filtern, Query-Parameter)

Gibt an, ib die Ausschnittdatensätze in den Ergebnissen dieser Query zurückgegeben werden.

Wert: `layout` oder `none`

- Wenn `-relatedsets.filter` auf `layout` gesetzt ist, wird die Einstellung **Anfangszeile** im FileMaker Pro-Dialogfeld „Ausschnitteinstellungen“ respektiert.
  - Wenn die Einstellung **Vertikaler Rollbalken** im Dialogfeld „Ausschnitteinstellungen“ aktiviert ist, verwenden Sie die Option `-relatedsets.max`, um die maximal zurückzugebende Anzahl an Datensätzen anzugeben. Siehe „–relatedsets.max (Ausschnittdatensätze beschränken, Query-Parameter)“ weiter unten.
  - Wennd ie Option **Vertikaler Rollbalken** deaktiviert ist oder die Option `-relatedsets.max` nicht verwendet wird, legt die Einstellung **Anzahl der Zeilen** im Dialogfeld „Ausschnitteinstellungen“ die Anzahl der zurückzugebenden Ausschnittdatensätze fest.
- Der Standardwert ist `none`, wenn dieser Parameter nicht angegeben ist. Wenn `-relatedsets.filter` auf `none` eingestellt ist, gibt die Web Publishing Engine alle Datensätze im Ausschnitt zurück. Die Werte für **Anfangszeile** und **Anzahl der Zeilen** aus dem Dialogfeld „Ausschnitteinstellungen“ wird ignoriert.

Notizen:

- Der Parameter `-relatedsets.filter` hat keine Auswirkung auf die Sortierung von Ausschnittdatensätzen bei XML-Abfragen. Die in FileMaker Pro definierte Sortierung wird beachtet, gleich, ob der Parameter `-relatedsets.filter` auf `layout` oder `none` gesetzt ist.
- Die Einstellung **Ausschnittdatensätze filtern** im Dialogfeld „Ausschnitteinstellungen“ wird für XML-Abfragen nicht unterstützt. Für die Einstellung **Ausschnittdatensätze filtern** angegebene Berechnungen werden ignoriert.

Optional bei: `-find`, `-edit`, `-new`, `-dup` und `-findquery`.

Beispiele:

```
http://localhost/fmi/xml/fmresultset.xml?-db=FMPHP_Sample&-lay=English
&-relatedsets.filter=none&-findany
```

```
http://localhost/fmi/xml/fmresultset.xml?-db=FMPHP_Sample
&-lay=English&relatedsets.filter=layout&-relatedsets.max=all&-findany
```

```
http://localhost/fmi/xml/fmresultset.xml?-db=FMPHP_Sample&-lay=English
&-relatedsets.filter=layout&-relatedsets.max=10&-findany
```

-relatedsets.max (Ausschnittdatensätze beschränken, Query-Parameter)

Gibt die maximale Anzahl an Ausschnittdatensätzen an, die in den Ergebnissen für diese Query zurückgegeben werden.

Wert: Eine Ganzzahl oder `all`.

- Der Parameter `-relatedsets.max` wird nur beachtet, wenn die Einstellung **Vertikaler Rollbalken** im FileMaker Pro-Dialogfeld „Ausschnitteinstellungen“ und der Parameter `-relatedsets.filter` auf `layout` gesetzt ist.
  - Wenn der Parameter `-relatedsets.max` eine Ganzzahl angibt, gibt die Web Publishing Engine die Anzahl an Ausschnittdatensätzen nach der ersten Zeile zurück.
  - Wenn der Parameter `-relatedsets.max all` angibt, gibt die Web Publishing Engine alle Ausschnittdatensätze zurück.

Hinweis Informationen zum Filtern von Ausschnittdatensätzen finden Sie unter „–relatedsets.filter (Ausschnittdatensätze filtern, Query-Parameter)“ oben.

Optional für: `-find`, `-edit`, `-new`, `-dup` und `-findquery`.

Beispiele:

```
http://localhost/fmi/xml/fmresultset.xml?-db=FMPHP_Sample
&-lay=English&relatedsets.filter=layout&-relatedsets.max=all&-findany
```

```
http://localhost/fmi/xml/fmresultset.xml?-db=FMPHP_Sample&-lay=English
&-relatedsets.filter=layout&-relatedsets.max=10&-findany
```

-script (Script) (Query-Parameter)

Gibt das FileMaker-Script an, das nach Ausführung des Query-Befehls und der Sortierung ablaufen soll. Weitere Informationen finden Sie unter „Erläuterung der Verarbeitung einer XML-Abfrage“ auf Seite 39

Wert: Scriptname.

Optional für: Alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script=myscript&-findall
```

Query-Parameter -script.param (Parameter an Script übergeben)

Übergibt einen Parameter an das durch `-script` angegebene FileMaker-Script.

Wert: Ein einzelner Textparameter.

- Um mehrere Parameter zu übergeben, können Sie einen String erstellen, der die Parameter begrenzt, und Ihr Script die einzelnen Parameter auslesen lassen. Übergeben Sie z. B. `param1|param2|param3` als Liste mit dem Zeichen `|`, das so kodiert ist: `param1%7Cparam2%7Cparam3`
- Um den Textparameter als Wert zu behandeln, der kein Text ist, kann Ihr Script den Textwert konvertieren. Um z. B. den Textwert in eine Zahl zu konvertieren, könnte Ihr Script Folgendes beinhalten: `LiesAlsZahl (Hole(ScriptParam))`
- Wenn Ihre Query `-script.param` ohne `-script` enthält, wird `-script.param` ignoriert.

- Wenn Ihre Query mehr als einen `-script.param` enthält, verwendet die Web Publishing Engine den letzten geparsten Wert.

Optional für: `-script`

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script=myscript&-script.param=Smith%7CChatterjee%7CSu
&-findall
```

Query-Parameter -script.prefind (Script vor Suchen)

Gibt das FileMaker-Script an, das vor dem Suchen und Sortieren von Datensätzen (falls angegeben) während der Verarbeitung des Query-Befehls `-find` ausgeführt wird.

Wert: Scriptname.

Optional für: Alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script.prefind=myscript&-findall
```

Query-Parameter -script.prefind.param (Parameter an Script vor Suchen übergeben)

Übergibt einen Parameter an das durch `-script.prefind` angegebene FileMaker-Script.

Wert: Ein einzelner Textparameter.

- Um mehrere Parameter zu übergeben, können Sie einen String erstellen, der die Parameter begrenzt, und Ihr Script die einzelnen Parameter auslesen lassen. Übergeben Sie z. B. `param1|param2|param3` als Liste mit dem Zeichen `|`, das so kodiert ist: `param1%7Cparam2%7Cparam3`
- Um den Textparameter als Wert zu behandeln, der kein Text ist, kann Ihr Script den Textwert konvertieren. Um z. B. den Textwert in eine Zahl zu konvertieren, könnte Ihr Script Folgendes beinhalten: `LiesAlsZahl (Hole(ScriptParam))`
- Wenn Ihre Query `-script.prefind.param` ohne `-script.prefind` enthält, wird `-script.prefind.param` ignoriert.
- Wenn Ihre Query mehr als einen `-script.prefind.param` enthält, verwendet die Web Publishing Engine den letzten geparsten Wert.

Optional für: `-script.prefind`

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script.prefind=myscript&-script.prefind.param=payroll
&-findall
```

Query-Parameter -script.presort (Script vor Sortieren)

Gibt das FileMaker-Script an, das nach dem Suchen von Datensätzen (falls angegeben) und vor dem Sortieren der Datensätze während der Verarbeitung des Query-Befehls `-find` ausgeführt wird.

Optional für: Alle Query-Befehle mit Ausnahme von `-dbnames`, `-layoutnames` und `-scriptnames`.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script.presort=myscript&-sortfield.1=dept
&-sortfield.2=rating&-findall
```

Query-Parameter -script.presort.param (Parameter an Script vor Sortieren übergeben)

Übergibt einen Parameter an das durch `-script.presort` angegebene FileMaker-Script.

Wert: Ein einzelner Textparameter.

- Um mehrere Parameter zu übergeben, können Sie einen String erstellen, der die Parameter begrenzt, und Ihr Script die einzelnen Parameter auslesen lassen. Übergeben Sie z. B. `param1|param2|param3` als Liste mit dem Zeichen `|`, das so kodiert ist: `param1%7Cparam2%7Cparam3`
- Um den Textparameter als Wert zu behandeln, der kein Text ist, kann Ihr Script den Textwert konvertieren. Um z. B. den Textwert in eine Zahl zu konvertieren, könnte Ihr Script Folgendes beinhalten: `LiesAlsZahl (Hole(ScriptParam))`
- Wenn Ihre Query `-script.presort.param` ohne `-script.presort` enthält, wird `-script.presort.param` ignoriert.
- Wenn Ihre Query mehr als einen `-script.presort.param` enthält, verwendet die Web Publishing Engine den letzten geparsten Wert.

Optional für: `-script.presort`

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-script.presort=myscript&-script.presort.param=18%7C65
&-sortfield.1=dept&-sortfield.2=rating&-findall
```

Query-Parameter -skip (Datensätze überspringen)

Gibt an, wie viele Datensätze in der Ergebnismenge übersprungen werden sollen.

Wert: Eine Zahl. Wenn der Wert größer als die Anzahl der Datensätze in der Ergebnismenge ist, wird kein Datensatz angezeigt. Der Standardwert beträgt 0.

Optional für: `-find` (Query-Befehl)

Im folgenden Beispiel werden die ersten 10 Datensätze in der Ergebnismenge ausgelassen und die Datensätze 11 bis 15 ausgegeben.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=departments&-skip=10&-max=5&-findall
```

Query-Parameter -sortfield (Sortierfeld)

Gibt das Feld an, das für das Sortieren verwendet wird.

Wert: Feldname

Optional für: `-find`- oder `-findall`-Query-Befehle

Der Query-Parameter `-sortfield` kann mehrfach verwendet werden, um mehrere Feldsortierungen vorzunehmen. Syntax für die Rangfolge der Sortierfelder:

```
-sortfield.precedence-number=fully-qualified-field-name
```

Dabei gilt: `rang-nummer` im Query-Parameter `-sortfield.rang-nummer` ist eine Zahl, die die Rangfolge für mehrere Sortierfelder angibt. Der Wert für `precedence-number`:

- muss bei 1 beginnen.
- muss sequenziell heraufgesetzt werden.
- darf nicht größer als 9 sein.

In folgendem Beispiel wird zuerst das Feld „dept“ sortiert und dann das Feld „rating“. Beide Felder werden aufsteigend sortiert, da der Query-Parameter `-sortorder` nicht angegeben wurde.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=performance&-sortfield.1=dept&-sortfield.2=rating&-findall
```

Query-Parameter -sortorder (Sortierfolge)

Gibt die Richtung einer Sortierung an.

Wert: Die Sortierfolge. Gültige Sortierfolgen (wobei `<wertelisten-name>` der Name einer Werteliste wie `Spezial` ist):

Schlüsselwort	Äquivalenter Operator in FileMaker Pro
`ascend`	Sortieren von a bis z, -10 bis 10
`descend`	Sortieren von z bis a, 10 bis -10
`<wertelisten-name>`	Sortieren nach der angegebenen Werteliste für das Feld im Layout

Optional für: `-find`- oder `-findall`-Query-Befehle

Erfordert: `-sortfield` (Query-Parameter)

Der Query-Parameter `-sortorder` kann mit dem Query-Parameter `-sortfield` verwendet werden, um die Sortierfolge mehrerer Sortierfelder anzugeben. Syntax für die Sortierfolge eines Sortierfelds:

```
-sortorder.precedence-number=sort-method
```

Dabei gilt:

- `precedence-number` im Query-Parameter `-sortorder.precedence-number` ist eine Zahl von 1 bis 9, die den Query-Parameter `-sortfield` angibt, auf den sich der Query-Parameter `-sortorder` bezieht.
- `sortier-methode` ist eines der Schlüsselwörter in obiger Tabelle für die Festlegung der Sortierfolge, z. B. „ascend“.

Im obigen Beispiel ist die Sortierfolge für das Sortierfeld mit der höchsten Rangnummer (dept) ascend und für das Sortierfeld mit der zweithöchsten Rangnummer (rating) descend. Die `precedence-number` 2 in `-sortorder.2` gibt an, dass der Query-Parameter `-sortorder.2=descend` sich auf den Query-Parameter `-sortfield.2=rating` bezieht.

Beispiel:

```
http://192.168.123.101/fmi/xml/fmresultset.xml?-db=employees
&-lay=performance&-sortfield.1=dept&-sortorder.1=ascend&-sortfield.2=rating
&-sortorder.2=descend&-findall
```

Hinweis Ein Sortierfeld, für das der Query-Parameter `-sortorder` nicht angegeben ist, verwendet standardmäßig aufsteigende Sortierung.

Anhang B
Fehlercodes für Custom Web Publishing

Die Web Publishing Engine generiert Fehlercodes für Datenbank- und Query-String-Fehler, die während einer XML-Daten-Abfrage auftreten können.

Dieser Anhang führt die Fehlercodes auf, die zum Zeitpunkt der Veröffentlichung dieses Dokuments bekannt waren. Eine Liste der neuesten Fehlercodes finden Sie in der FileMaker Knowledge Base (http://help.filemaker.com).

Fehlercodenummern in XML-Format

Die Web Publishing Engine generiert für eine in XML-Format veröffentlichte Datenbank jedes Mal einen Fehlercode, wenn Daten angefordert werden. Diese Art von Fehlercodewert wird am Beginn des XML-Dokuments in das `<error code>`-Element für die `fmresultset`-Grammatik bzw. im `<ERRORCODE>`-Element für die Grammatik `FMPXMLRESULT` oder `FMPXMLLAYOUT` eingefügt. Der Fehlercode 0 gibt an, dass kein Fehler aufgetreten ist.

Beispiel des Datenbankfehlercodes in der `fmresultset`-Grammatik:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE fmresultset PUBLIC "-//FMI//DTD fmresultset//EN"
"/fmi/xml/fmresultset.dtd">
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version="1.0">
    <error code="0"></error>
```

Beispiel des Datenbankfehlercodes in der `FMPXMLRESULT`-Grammatik:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE FMPXMLRESULT PUBLIC "-//FMI//DTD FMPXMLRESULT//EN"
"/fmi/xml/FMPXMLRESULT.dtd">
<fmpxmlresult xmlns="http://www.filemaker.com/fmpxmlresult">
    <ERRORCODE>0</ERRORCODE>
```

Als Entwickler der Custom Web Publishing-Lösung bleibt es Ihnen überlassen, den Wert des `<error code>`- bzw. `<ERRORCODE>`-Elements zu prüfen und geeignete Maßnahmen vorzusehen. Die Web Publishing Engine behandelt keine Datenbankfehler.

Fehlercodenummern für FileMaker-Datenbanken

Fehlernummer	Beschreibung
-1	Unbekannter Fehler
0	Kein Fehler
1	Aktion durch Benutzer abgebrochen
2	Speicherfehler
3	Befehl nicht verfügbar (z. B. falsches Betriebssystem, falscher Modus etc.)
4	Befehl unbekannt
5	Befehl ungültig (z. B. ist für den Scriptschritt „Feldwert setzen“ keine Formel angegeben)
6	Datei ist schreibgeschützt

Fehlernummer	Beschreibung
7	Speicherüberlauf
8	Leeres Ergebnis
9	Ungenügende Berechtigungen
10	Angeforderte Daten fehlen
11	Name ist nicht gültig
12	Name existiert bereits
13	Datei oder Objekt ist in Gebrauch
14	Außerhalb des gültigen Bereichs
15	Teilen durch null nicht möglich
16	Operation fehlgeschlagen, Anfrage wiederholen (z. B. eine Benutzeranfrage)
17	Konvertierung von fremdem Zeichensatz in UTF-16 fehlgeschlagen
18	Client muss Kontoinformationen liefern, um fortzufahren
19	Zeichenfolge enthält andere Zeichen als A-Z, a-z, 0-9 (ASCII)
20	Befehl oder Vorgang durch ausgelöstes Script abgebrochen
100	Datei fehlt
101	Datensatz fehlt
102	Feld fehlt
103	Beziehung fehlt
104	Script fehlt
105	Layout fehlt
106	Tabelle fehlt
107	Index fehlt
108	Werteliste nicht vorhanden
109	Berechtigung fehlt
110	Bezugstabellen fehlen
111	Feldwiederholung ist ungültig
112	Fenster fehlt
113	Funktion fehlt
114	Dateiverweis fehlt
115	Menüset fehlt
116	Layoutobjekt fehlt
117	Datenquelle fehlt
118	Thema fehlt
130	Dateien sind beschädigt oder fehlen und müssen neu installiert werden
131	Dateien des Sprachpakets fehlen (z. B. Vorlagendateien)
200	Zugriff auf Datensatz verweigert
201	Feld kann nicht geändert werden
202	Zugriff auf Feld verweigert

Fehlernummer	Beschreibung
203	Keine zu druckenden Datensätze in der Datei bzw. Passwort erlaubt kein Drucken
204	Kein Zugriff auf Feld(er) in Sortierfolge
205	Benutzer hat keine Zugriffsrechte, um neue Datensätze zu erstellen; Import überschreibt bestehende Daten
206	Benutzer hat keine Zugriffsrechte, um das Passwort zu ändern, oder die Datei ist schreibgeschützt
207	Benutzer hat nicht genügend Zugriffsrechte, um das Datenbankschema zu ändern, oder die Datei ist schreibgeschützt
208	Passwort enthält zu wenige Zeichen
209	Neues Passwort muss sich vom bestehenden unterscheiden
210	Benutzerkonto ist inaktiv
211	Passwort ist abgelaufen
212	Ungültiges Benutzerkonto und/oder Passwort. Versuchen Sie es erneut
213	Benutzerkonto und/oder Passwort existieren nicht
214	Zu viele Anmeldeversuche
215	Administratorrechte können nicht dupliziert werden
216	Gastkonto kann nicht dupliziert werden
217	Benutzer hat nicht genügend Zugriffsrechte, um Administratorkonto zu ändern
218	Passwort und Passwortüberprüfung stimmen nicht überein
300	Datei ist geschützt oder in Gebrauch
301	Datei ist blockiert durch anderen Anwender
302	Tabelle ist blockiert durch anderen Anwender
303	Datenbankschema ist blockiert durch anderen Anwender
304	Layout ist blockiert durch anderen Anwender
306	Datensatzänderungs-ID stimmt nicht überein
307	Transaktion konnte wegen eines Kommunikationsfehlers mit dem Host nicht gesperrt werden
308	Thema ist blockiert durch anderen Anwender
400	Suchkriterien sind leer
401	Kein Datensatz entspricht der Abfrage
402	Kein Abgleichsfeld für eine Referenz
403	Maximales Datensatzlimit für FileMaker Pro-Demo wird überschritten
404	Ungültige Sortierfolge
405	Angegebene Datensatzzahl übersteigt die Anzahl der ausschließbaren Datensätze
406	Ungültige Kriterien für Ersetzen/Neunummerierung
407	Ein oder beide Gruppierfeld(er) fehlen (ungültige Beziehung)
408	Angegebenes Feld hat ein Datenformat, das diesem Befehl nicht entspricht
409	Ungültige Importfolge
410	Ungültige Exportfolge
412	Falsche Version von FileMaker Pro verwendet, um die Datei wiederherzustellen
413	Angegebenes Feld hat ungeeigneten Feldtyp

Fehlernummer	Beschreibung
414	Layout kann das Ergebnis nicht anzeigen
415	Ein oder mehrere erforderliche Bezugsdatensätze sind nicht verfügbar
416	Primärschlüssel erforderlich für Datenquelltabelle
417	Die Datenbank ist keine unterstützte Datenquelle
500	Datumswert entspricht nicht den Überprüfungskriterien
501	Zeitwert entspricht nicht den Überprüfungskriterien
502	Zahlenwert entspricht nicht den Überprüfungskriterien
503	Feldwert entspricht nicht der Bereichsüberprüfung
504	Feldwert entspricht nicht der Eindeutigkeitsüberprüfung
505	Feldwert existiert nicht in der Datenbank und entspricht nicht der Existenzüberprüfung
506	Feldwert entspricht nicht der Überprüfung nach Bestandteil einer Werteliste
507	Feldwert entspricht nicht der Überprüfung durch Berechnung
508	Ungültiger Wert wurde in Suchenmodus eingegeben
509	Feld verlangt gültigen Wert
510	Bezugswert ist leer oder nicht verfügbar
511	Der Wert im Feld überschreitet die maximale Anzahl der zulässigen Zeichen
512	Datensatz wurde bereits durch anderen Anwender geändert
600	Druckerfehler aufgetreten
601	Kombination von Kopf- und Fußbereich übersteigt eine Seitenlänge
602	Datenbereich passt für aktuelle Spalteneinstellung nicht auf eine Seite
603	Verbindung zum Drucker getrennt
700	Datei hat falschen Dateityp für Import
706	EPSF-Datei hat keine Übersichtsgrafik
707	Grafikfilter nicht vorhanden
708	Dateiimport nicht möglich bzw. Farbmonitor für Import erforderlich
709	Import des QuickTime-Films fehlgeschlagen
710	QuickTime-Dateiverweis konnte nicht aktualisiert werden, da für Datenbankdatei nur Lesezugriff besteht
711	Importfilter nicht vorhanden
714	Zugriffsrechte reichen für diesen Befehl nicht aus
715	Benannter Bereich oder Tabellenblatt von Excel konnte nicht gefunden werden
716	Eine SQL Anfrage mit DELETE, INSERT oder UPDATE ist nicht für ODBC Import erlaubt
717	Zum Fortsetzen des Imports bzw. Exports sind nicht genügend XML/XSL-Daten vorhanden
718	XML-Parsingfehler (von Xerces)
719	Fehler beim Transformieren von XML mit XSL (von Xalan)
720	Fehler beim Export: Das vorgesehene Format unterstützt keine Wiederholfelder
721	Im Parser oder Transformer ist ein unbekannter Fehler aufgetreten
722	Daten können nicht in eine Datei importiert werden, die keine Felder hat
723	Sie sind nicht berechtigt, Datensätze in der Zieltabelle hinzuzufügen oder zu ändern

Fehlernummer	Beschreibung
724	Sie sind nicht berechtigt, Datensätze in der Zieltabelle hinzuzufügen
725	Sie sind nicht berechtigt, Datensätze in der Zieltabelle zu ändern
726	In der Importdatei sind mehr Datensätze vorhanden als in der Zieltabelle. Nicht alle Datensätze werden importiert
727	In der Zieltabelle sind mehr Datensätze vorhanden als in der Importdatei. Nicht alle Datensätze werden aktualisiert
729	Fehler beim Import. Datensätze konnten nicht importiert werden
730	Nicht unterstützte Excel-Version (Datei in Excel 2000-Format oder neuere, unterstützte Version konvertieren und erneut probieren)
731	Die für den Import ausgewählte Datei enthält keine Daten
732	Diese Datei kann nicht eingefügt werden, weil sie selbst weitere Dateien enthält
733	Eine Tabelle kann nicht in sich selbst importiert werden
734	Dieser Dateityp kann nicht als Bild dargestellt werden
735	Dieser Dateityp kann nicht als Bild dargestellt werden. Er wird eingefügt und dann als Datei dargestellt
736	Zu viele Daten für den Export in dieses Format. Sie werden abgeschnitten
737	Die Bento-Tabelle, die Sie importieren, fehlt
800	Datei konnte auf Datenträger nicht erstellt werden
801	Temporärdatei konnte auf Systemdatenträger nicht erstellt werden
802	Datei konnte nicht geöffnet werden Dieser Fehler kann durch eine oder mehrere der folgenden Möglichkeiten verursacht werden: ▪ Ungültiger Datenbankname ▪ Datei ist in FileMaker Server geschlossen ▪ Ungültige Berechtigung
803	Datei in Einzelbenutzer-Status oder Host nicht vorhanden
804	Datei konnte in ihrem aktuellen Status nicht mit Nur-Lese-Zugriff geöffnet werden
805	Datei ist beschädigt; stellen Sie die Datei wieder her
806	Datei kann mit dieser Version von FileMaker Pro nicht geöffnet werden
807	Datei ist keine FileMaker Pro-Datei oder ist schwer beschädigt
808	Datei kann wegen beschädigter Zugriffsrechte nicht geöffnet werden
809	Datenträger voll
810	Datenträger fixiert
811	Temporärdatei kann nicht als FileMaker Pro-Datei geöffnet werden
813	Fehler bei Datensatz-Synchronisation im Netzwerk
814	Datei(en) kann (können) nicht geöffnet werden, da die maximale Anzahl an Dateien geöffnet ist
815	Referenzdatei konnte nicht geöffnet werden
816	Datei konnte nicht konvertiert werden
817	Der Bindungsschlüssel der Datei stimmt nicht mit dieser Laufzeitanwendung überein
819	Lokale Kopie einer remote Datei kann nicht gespeichert werden
820	Datei wird geschlossen
821	Host hat die Verbindung getrennt

Fehlernummer	Beschreibung
822	FMI-Dateien nicht gefunden; fehlende Dateien neu installieren
823	Datei kann nicht auf Einzelbenutzer gesetzt werden, da Gäste verbunden sind
824	Datei ist beschädigt oder keine FileMaker-Datei
825	Datei ist nicht autorisiert, die geschützte Datei zu referenzieren
826	Der angegebene Dateipfad ist kein gültiger Dateipfad
850	Der Pfad ist für das Betriebssystem ungültig
851	Löschen einer externen Datei vom Datenträger nicht möglich
852	Schreiben einer Datei auf den externen Speicher nicht möglich
900	Allgemeiner Fehler in der Rechtschreibprüfung
901	Standardwörterbuch nicht installiert
902	Hilfe-System konnte nicht gestartet werden
903	Befehl kann nicht in einer gemeinsam genutzten Datei verwendet werden
904	Befehl kann nur in einer Datei verwendet werden, die von FileMaker Pro Server freigegeben wurde
905	Kein aktives Feld ausgewählt, Befehl kann nur mit aktivem Feld verwendet werden
906	Die aktuelle Datei wird nicht bereitgestellt. Der Befehl kann nur verwendet werden, wenn die Datei bereitgestellt wird
920	Initialisierung der Rechtschreibprüfung nicht möglich
921	Anwenderwörterbuch kann nicht zur Bearbeitung geladen werden
922	Anwenderwörterbuch existiert nicht
923	Anwenderwörterbuch ist schreibgeschützt
951	Ein unerwarteter Fehler ist aufgetreten
954	Nicht unterstützte XML-Grammatik
955	Kein Datenbankname
956	Maximale Anzahl von Datenbanksitzungen überschritten
957	Widersprüchliche Befehle
958	Parameter fehlt in Query
959	Custom Web Publishing ist deaktiviert
960	Parameter ungültig
1200	Generischer Rechenfehler
1201	In dieser Funktion gibt es zu wenige Parameter
1202	In dieser Funktion gibt es zu viele Parameter
1203	Unerwartetes Ende der Berechnung
1204	Es wird eine Zahl, eine Textkonstante, ein Feldname bzw. „(“ erwartet
1205	Kommentar ist nicht mit „*/“ beendet
1206	Textkonstante muss mit einem Anführungszeichen enden
1207	Klammer unvollständig
1208	Operator fehlt, Funktion nicht gefunden oder „(“ nicht erwartet
1209	Name (z. B. Feldname oder Layoutname) fehlt
1210	Plugin-Funktion wurde bereits registriert

Fehlernummer	Beschreibung
1211	In dieser Funktion ist die Listennutzung nicht zulässig
1212	Hier wird ein Operator (z. B. +, -, *) erwartet
1213	Diese Variable wurde bereits in der SetzeVars-Funktion definiert
1214	MITTELWERT, ANZAHL, ERWEITERN, HOLEWIEDERHOLFELDWERT, MAX, MIN, NBW, STABW, SUMME und HOLEERGEBNISWERT: Ausdruck gefunden, wo nur ein Feld benötigt wird
1215	Dieser Parameter ist ein ungültiger Statusfunktionsparameter
1216	Als erstes Argument einer HOLEERGEBNISWERT-Funktion können nur Statistikfelder angegeben werden
1217	Gruppierfeld ist ungültig
1218	Zahl kann nicht berechnet werden
1219	Ein Feld kann nicht in seiner eigenen Formel benutzt werden
1220	Feldtyp muss normal oder berechnet sein
1221	Datentyp muss Zahl, Datum, Zeit oder Zeitstempel sein
1222	Formel kann nicht gespeichert werden
1223	Die angegebene Funktion ist noch nicht implementiert
1224	Die angegebene Funktion existiert nicht
1225	Die angegebene Funktion wird in diesem Kontext nicht unterstützt
1300	Der angegebene Name kann nicht verwendet werden
1400	ODBC-Client-Treiberinitialisierung fehlgeschlagen. Stellen Sie sicher, dass die ODBC-Client-Treiber richtig installiert sind
1401	Umgebung konnte nicht zugeteilt werden (ODBC)
1402	Umgebung konnte nicht freigegeben werden (ODBC)
1403	Verbindung trennen fehlerhaft (ODBC)
1404	Verbindung konnte nicht zugeteilt werden (ODBC)
1405	Verbindung konnte nicht freigegeben werden (ODBC)
1406	Überprüfung von SQL API fehlgeschlagen (ODBC)
1407	Anweisung konnte nicht zugeteilt werden (ODBC)
1408	Erweiterter Fehler (ODBC)
1409	Erweiterter Fehler (ODBC)
1410	Erweiterter Fehler (ODBC)
1411	Erweiterter Fehler (ODBC)
1412	Erweiterter Fehler (ODBC)
1413	Erweiterter Fehler (ODBC)
1414	SQL-Anweisung zu lang
1450	Aktion erfordert PHP-Berechtigungserweiterung
1451	Aktion erfordert, dass die aktuelle Datei remote ist
1501	SMTP-Authentifizierung fehlgeschlagen
1502	Verbindung vom SMTP-Server abgelehnt
1503	SSL-Fehler

Fehlernummer	Beschreibung
1504	Der SMTP-Server erfordert eine verschlüsselte Verbindung
1505	Die angegebene Authentifizierung wird vom SMTP-Server nicht unterstützt
1506	E-Mail(s) konnte(n) nicht erfolgreich versendet werden
1507	Anmeldung am SMTP-Server nicht möglich
1550	Plugin konnte nicht geladen werden oder Plugin ist kein gültiges Plugin
1551	Plugin kann nicht installiert werden. Ein bestehendes Plugin kann nicht gelöscht werden oder Schreiben in den Ordner oder Datenträger nicht möglich
1626	Protokoll wird nicht unterstützt
1627	Authentifizierung fehlgeschlagen
1628	Es ist ein SSL-Fehler aufgetreten
1629	Verbindungs-Zeitfehler; das Sitzungszeitlimit beträgt 60 Sekunden
1630	URL-Format nicht korrekt
1631	Verbindung fehlgeschlagen

Anhang C
Änderungen bei XML-Queries in FileMaker 12

Dieser Anhang führt die syntaktischen und semantischen Änderungen an XML-Query-Befehlen und -Parametern für FileMaker 12 auf.

XML-Query-Änderungen in der Syntax

Die folgenden Query-Befehle wurden in FileMaker 12 gelöscht:

- `-process`

Die folgenden Query-Parameter wurden in FileMaker 12 gelöscht:

- `-encoding`
- `-grammar`
- `-stylehref`
- `-styletype`
- `-token.[Zeichenfolge]`

XML-Query-Änderungen in der Semantik

Unterschiede im Query-Parsing

Die Web Publishing Engine für FileMaker 12 wartet mit erheblichen Änderungen an der Reihenfolge auf, in der Query-Befehle analysiert werden. Weitere Informationen zur Parsing-Reihenfolge von Query-Befehlen finden Sie unter „Query-Befehls-Parsing“ auf Seite 48.

Beispielsweise unterstützt FileMaker 12 resultierend aus der Änderung der Parsing-Reihenfolge von Query-Befehlen eine `-find`-Query ohne Suchkriterien nicht mehr, auch wenn Sie einen `-script`-Parameter einschließen, der das Suchkriterium angibt. Zum Beispiel würde der folgende Query-Befehl in FileMaker 11 funktionieren, vorausgesetzt, dass das Script `SucheAusfuehren` die Suchkriterien angibt:

```
-db=Kunden&-lay=Konto&-script=SucheAusfuehren&-find
```

Aber in FileMaker 12 gibt dieses Beispiel Fehlercode 400 (Suchkriterien sind leer) zurück. In FileMaker 12 können Sie `-findall` als Suchkriterium einschließen:

```
-db=Kunden&-lay=Konto&-findall&-script=SucheAusfuehren
```

Zusätzlich gilt in Bezug auf die Web Publishing Engine für FileMaker 12 Folgendes:

- Sie ignoriert gültige Befehle oder Parameter, die irrelevant in einer Query sind. Zum Beispiel ist in der folgenden Query der Parameter `-lop` ein gültiger Parameter, aber im Befehl `-delete` irrelevant:

  ```
  -db=test&-lay=test&-recid=82&-delete&f1=hi&-lop=and
  ```

  Da der Parameter `-lop` das gültige Argument `and` verwendet, wird er ignoriert und kein Fehler zurückgegeben.

- Sie ignoriert den gleichen Befehl, wenn er mehr als einmal weitergegeben wird. Beispiel: `-dbnames&-dbnames`

  Die Web Publishing Engine gibt jedoch einen Fehler zurück, wenn Sie mehr als zwei unterschiedliche Befehle in der gleichen Query angeben. Beispiel: `-find&-edit`
- Sie ignoriert doppelte kaufmännische Unds bzw. leere Parameter in einer Query-Anfrage. FileMaker 11 gibt einen Fehler zurück.
- Sie konvertiert alle reservierten Wörter in Kleinbuchstaben. FileMaker 11 gibt in bestimmten Fällen Fehler für großgeschriebene reservierte Wörter zurück.
- Sie ignoriert den Parameter `field.op` für eine Query, die keine `-find`-Query ist. FileMaker 11 ersetzt das Feld durch den Parameter `field.op`.
- Sie ignoriert alle nicht globalen Felder, die für die Befehle `-findany`, `-findall`, `-find` mit recid, `-findquery`, `-delete` und `-dup` weitergegeben werden.
- Sie verlangt, dass globalen Feldern das Suffix „`.global`“ angehängt wird.

Unterschiede in der Query-Verarbeitung

Für die Web Publishing Engine für FileMaker 12 gilt Folgendes:

- Sie bearbeitet globale Felder für die Befehle `-findall`, `-findany`, `-find`, `-findquery`, `-new`, `-edit` und `-dup`.
- Sie setzt globale Felder zurück, die manuell auf ihre Originalwerte zurückgesetzt wurden, nachdem eine Query verarbeitet und Ergebnisse zurückgegeben wurden. Globale Felder, die über ein Script bearbeitet werden, werden nicht zurückgesetzt.
- Sie gibt alle Datensätze zurück, die den Suchkriterien für eine `-find`-Query mit `field-name.op` gleich `neq` und `-lop` gleich `or` entsprechen. FileMaker 11 verarbeitet diese Query nicht richtig.
- Sie gibt alle Datensätze zurück, die einen Wert für jeden Feldparameter haben, der als leere Zeichenfolge für eine `-find`-Query weitergegeben wird. FileMaker 11 entfernt jedes Feld mit leerer Zeichenfolge aus den Suchkriterien.
- Sie gibt Dezimalsekunden für Datums-, Zeit- und Zeitstempelfelder zurück. FileMaker 11 gibt Integersekunden zurück.
- Für Wertelisten, die mit der Einstellung **Nur Bezugswerte aufnehmen, beginnend mit** definiert wurden, gibt sie keinen Wert zurück, wenn keine recid angegeben ist. FileMaker 11 gibt die Werte in Bezug auf den ersten Datensatz zurück.

Unterschiede in den zurückgegebenen Fehlercodes

Fehlerbedingung	FileMaker 11-Fehlercode	FileMaker 12-Fehlercode
Ungültige max- oder skip-Werte. Beispiel: `-db=basissuchen&-lay=layouteins&-max=-1&-findall`	5 (Befehl ungültig)	960 (Parameter ungültig)
Sortierfolgenummer fehlt oder ungültig. Beispiel: `-db=basissortierungen&-lay=layouteins &-sortfield.=textfeld&-findall`	5 (Befehl ungültig)	404 (Ungültige Sortierfolge)
Ungültige Bezugstabelle angegeben. Beispiel: `-db=relsuchen &-lay=layouteins &aliasexistiertnicht::bezugstextfeld=eintext&-find`	102 (Feld fehlt)	106 (Tabelle fehlt)

Fehlerbedingung	FileMaker 11-Fehlercode	FileMaker 12-Fehlercode
Ein leeres Sortierfeld in einer Query. Beispiel: `-db=basissortierungen&-lay=layouteins&-sortfield.1=&-findall`	Ignoriert. Kein Fehlercode.	102 (Feld fehlt)
–find-Query ohne jegliche Felder abgesetzt. Beispiel: `-db=basisbearbeitung&-lay=layouteins&-find`	Kein Fehler. FileMaker 11 gibt alle Datensätze zurück.	400 (Suchkriterien sind leer)
–find-Query nur mit globalen Feldern abgesetzt. Beispiel: `-db=basissuchen&-lay=layouteins &globalestextfeld.global=umsatz&-find`	Kein Fehler. FileMaker 11 gibt alle Datensätze zurück.	400 (Suchkriterien sind leer)

Index

A

Abfragen von Ausschnittfeldern 51
Abfragen von XML-Daten 24
Abrufen verfügbarer Scriptnamen 54
Admin Console 15, 24
Aktivieren von Instant Web Publishing in Datenbank 14
ASCII-Zeichen, in XML-Dokumenten 37
Attribut für automatische Eingabe 30
Attribut für vierstellige Jahreszahlen 30
Ausschnitte
  Anfangszeile 61
  Anzahl der Datensätze 61
  Datensätze bearbeiten 50
  Datensätze hinzufügen 50
  Datensätze löschen 51
  Layout 61
  Sortieren von Datensätzen 61
Ausschnittfeld-Queries 61, 62
Authentifizierung von Web-Benutzern 14
Authentifizierung Web-Benutzer 14

B

Befehle, Query *s.* Query-Strings
Begrenzen von Ausschnittfelddatensätzen 62
Beispiele für
  Generierte FMPXMLLAYOUT-Grammatik 36
  Generierte FMPXMLRESULT-Grammatik 33
  Generierte fmresultset-Grammatik 31
Benutzernamen
  Authentifizierung Web-Benutzer 14
  Für Custom Web Publishing definieren 14
Berechtigung, Zuweisen für Custom Web Publishing 14

C

Compound-Suche (Query-Befehl) 53
Compound-Suche (Query-Parameter) 59
Containerfelder
  Inhalt veröffentlichen 16
  Mit extern gespeicherten Daten 17
  Mit referenzierten Dateien 16
  URL-Syntax für Zugriff in XML-Lösungen 25
  Wie Web-Benutzer auf Daten zugreifen 18
Custom Web Publishing
  Definition 7
  Erweitertes Zugriffsrecht für 14
  Gastkonto 15
  In Datenbank aktivieren 14
  In Web Publishing Engine aktivieren 15
  Mit PHP 9
  Mit XML 9, 22
  Neue Funktionen in 11
  Scripts 20
  Scripts verwenden 18
  Statische IP-Adresse verwenden 13
  Übersicht 7
  Voraussetzungen für das 12
  Web-Benutzer-Zugr. auf Lösungen 14
  Zugriff von IP-Adressen in Web-Server beschränken 15
Custom Web Publishing Engine (CWPE) 23

D

Element <datasource> 29
Datenbanken, veröffentlichte schützen 15
Datenbankfehlercodes 28
–db (Query-Parameter) 55
–dbnames (Query-Befehl) 52
–delete (Query-Befehl) 52
–delete.related query, Parameter 51
Document Type Definitions (DTDs) 28, 32
Dokumentation 6
–dup (Query-Befehl) 52

E

–edit (Query-Befehl) 52
Elektronische Dokumentation 6
Elemente
  Datenbankfehlercodes 28
  In FMPXMLLAYOUT-Grammatik 34
  In FMPXMLRESULT-Grammatik 32
  In fmresultset-Grammatik 29
–encoding (Query-Parameter)
  Gelöscht in FileMaker 12 75
Erneut anmelden, Script 15
<error code-> und <ERRORCODE>-Elemente 67
Erstellen eines neuen Datensatzes 54
Erweitertes Zugriffsrecht für Custom Web Publishing 14
Exportieren von XML-Daten 22
Extensible Markup Language (XML) *Siehe* XML

F

Fehler
  Datenbankfehlercode-Elemente 28
  Datenbankfehlercodes 67
  Erläuterung von Fehlercodes 67
  Protokolldateien für Web-Server 43
Fehlerbehebung
  Custom Web Publishing-Webseiten 42
  XML-Dokumentzugriff 40
Feldnamen, voll qualifizierte Syntax 49
–field (Query-Parameter, für Containerfeld) 56
<field-definition> Element 30
fieldname (Query-Parameter, nicht für Containerfeld) 56
–fieldname.op (Query-Parameter) 57
FileMaker API for PHP 9
  Definition 9

FileMaker Pro, Unterschied zu Web Publishing Engine 22
FileMaker Server
  Dokumentation 6
  Installieren 6
FileMaker Server Admin Console 15, 24
Filtern von Ausschnittfelddatensätzen 61
–find (Query-Befehl) 53
–findall (Query-Befehl) 53
–findany (Query-Befehl) 53
-findquery (Query-Befehl) 53
FMPXMLLAYOUT-Grammatik 22, 34–36
  Im Vergleich mit anderen Grammatiken 27
FMPXMLRESULT-Grammatik 22, 32–33
  Im Vergleich mit anderen Grammatiken 27
fmresultset-Grammatik 22, 29–31
  Im Vergleich mit anderen Grammatiken 27
fmxml (Schlüsselwort zur Aktivierung von XML-Publishing) 14, 24

G

Gastkonto
  Aktivieren 15
  Deaktivieren 15
  Mit Custom Web Publishing 15
Gelöscht in FileMaker 12
  –encoding (Query-Parameter) 75
  –grammar (Query-Parameter) 75
  –process (Query-Befehl) 75
  –stylehref (Query-Parameter) 75
  –styletype (Query-Parameter) 75
  –token (Query-Parameter) 75
Globales Attribut 30
–grammar (Query-Parameter)
  Gelöscht in FileMaker 12 75
Grammatiken für XML, Beschreibung 27

H

Hinweise zur Dokumentation 6, 13
HTML
  Formulare für XML-Abfragen 24

I

Importieren von XML-Daten 22
Installationsdokumentation 6
Instant Web Publishing
  Definition 7
  Dokumentation 6

J

JDBC-Dokumentation 6

K

Kodierung
  URL 26
  XML-Daten 27, 37
Konten und Zugriffsrechte
  Für Custom Web Publishing aktivieren 14
  Gastkonto 15
  Scripts 19

L

–lay (Query-Parameter) 39, 58
–lay.response (Query-Parameter) 39, 58
Layout bei Abfrage von XML-Daten angeben 39
Layout wechseln für XML-Antwort 39
Layoutinformationen abrufen 55
Layoutnamen abrufen 54
–layoutnames (Query-Befehl) 54
Layouts, für XML-Antwort wechseln 39
–lop (Query-Parameter) 58
Löschen von Ausschnittdatensätzen 51

M

–max (Query-Parameter) 58
max-characters, Attribut 30
max-repeat, Attribut 30
Element <metadata> 30
MIME (Multipurpose Internet Mail Extensions-)-Typen 16
–modid (Query-Parameter) 59

N

name, Attribut 30
Namespaces für
  XML 27
Neue Funktionen in Custom Web Publishing 11
–new (Query-Befehl) 54
not-empty, Attribut 30
numeric-only, Attribut 30
Nummern für
  Datenbankfehlercodes 67

O

ODBC-Dokumentation 6
Online-Dokumentation 6
Operatoren, Vergleich 57

P

Parameter, Query *s.* Query-Strings
Passwort ändern (Script) 15
Passwörter
  Authentifizierung Web-Benutzer 14
  Für Custom Web Publishing definieren 14
  Kein Anmeldepasswort 15
  Passwort ändern (Script) 15
PDFs 6
PHP
  Vorteile 9
PHP API für Custom Web Publishing 9

–process (Query-Befehl)
Gelöscht in FileMaker 12 75
Programmprotokoll 44
progressiven Download 16, 18
Protokolldateien 42
Beschreibung 43
Tomcat 46
web_server_module_log.txt 45
Web-Server-Zugriff 43

Q

–query (Query-Parameter) 59
Query-Strings 37, 47
Befehle und Parameter 37, 47
Datensätze in Ausschnitte einfügen 50
Datensätze in Ausschnitten bearbeiten 50
Richtlinien für 47
Variablenfeld, Syntax 51
Voll qualifizierter Feldname, Syntax 49
XML 37, 47
QuickTime-Filme, im Web veröffentlichen 17

R

–recid (Query-Parameter) 60
Reihenfolge der XML-Abfrageverarbeitung 39
<relatedset-definition> Element 30
–relatedsets.filter (Query-Parameter) 61
–relatedsets.max (Query-Parameter) 62
result, Attribut 30
Element <resultset> 30

S

SAT
Siehe FileMaker Server Admin Console 15
Schlüsselwörter zur Aktivierung von Custom Web Publishing 14, 24
Schützen veröffentlichter Datenbanken 15
–script (Query-Parameter) 62
–script.param (Query-Parameter) 62
–script.prefind (Query-Parameter) 63
–script.prefind.param (Query-Parameter) 63
–script.presort (Query-Parameter) 63
–script.presort.param (Query-Parameter) 64
–scriptnames (Query-Befehl) 54
Scripts
Erneut anmelden 15
Für XML-Abfragen 24
In Custom Web Publishing 18
Konten und Zugriffsrechte 19
Passwort ändern 15
Tipps und Überlegungen 19
Trigger 21
Sicherheit
Dokumentation 8
Konten und Passwörter 15
Richtlinien zum Schützen veröffentlichter Datenbanken 15
Zugriff von IP-Adressen beschränken 15
–skip (Query-Parameter) 64
–sortfield (Query-Parameter) 64
Sortieren von Ausschnittfelddatensätzen 61
–sortorder (Query-Parameter) 65
SSL (Secure Sockets Layer)-Verschlüsselung 16
Statisches Veröffentlichen, Definition 7
–stylehref (Query-Parameter)
Gelöscht in FileMaker 12 75
Stylesheets
Testen 42
–styletype (Query-Parameter)
Gelöscht in FileMaker 12 75

T

Testen
Webseiten 42
XML-Ausgabe 42
Textkodierung
Generierte XML-Daten 27
URL 26
time-of-day, Attribut 30
–token (Query-Parameter)
Gelöscht in FileMaker 12 75
Tomcat
Verwenden von Protokolldateien 46
Trigger 21
type, Attribut 30

U

Überblick über Schritte für
XML-Datenzugriff 24
Übersicht
Custom Web Publishing 7
Überwachen von Webseiten 43
Unicode-Zeichen 37
URL-Syntax für
Containerobjekte in XML-Lösungen 25
XML-Abfragen 25
URL-Textkodierung 26
UTF-8 (Unicode Transformation 8 Bit)
Format 26, 37

V

Variablenfelder
Syntax 51
Verarbeitung einer Web Publishing Engine-Anforderung 8
Verfügbare Datenbanklayouts 54
Verfügbare Scripts 54
Vergleichsoperatoren für Felder 57
Veröffentlichen im Web
Containerfeldobjekte 16
Datenbankfehlercodes 67
QuickTime-Filme 17
Schützen von Datenbanken 15
Verbindung zum Internet oder Intranet 13
Voraussetzungen für das 12

XML verwenden 24
–view (Query-Befehl) 55
Voll qualifizierter fieldname, Syntax 49
Voraussetzungen für Custom Web Publishing 12

W

Web Publishing Core
Illustration 23
Web Publishing Engine
Admin Console 24
Anforderungsverarbeitung 8
Beschreibung 8
Erzeugte Fehlercodes 67
Programmprotokoll 44
Vorteile 11
XML-Daten generieren 23
XML-Dokumente erzeugen 24
web_server_module_log.txt (Protokolldatei) 45
Web-Benutzer
Containerfelddaten verwenden 18
Voraussetzungen für Zugriff auf Custom Web Publishing-Lösungen 12
Zugriff auf geschützte Datenbanken 14
Webbrowser
Rolle in XML-Abfragen 23
Web-Ordner, Containerfeldobjekte kopieren 16
Webseiten
FileMaker Support-Seiten 6
Mit FileMaker Web Publishing Engine erstellen 11
Testen 42
Überwachen 43
Web-Server
MIME-Typenunterstützung 16
Protokolldateien 43
Rolle in XML-Abfragen 23
Wechseln des Layouts für eine XML-Antwort 39

X

XML
Beschreibung 22
Daten abfragen 24
Document Type Definitions (DTDs) 28, 29, 32
Fehlerbehebung beim Zugriff auf XML-Dokumente 40
FMPXMLLAYOUT-Grammatik 34
FMPXMLRESULT-Grammatik 32
fmresultset-Grammatik 29
<field-definition> Element 30
<relatedset-definition> Element 30
Element <datasource> 29
Element <metadata> 30
Element <resultset> 30
Grammatiken, Beschreibung 27
Im UTF-8-Format kodiert 27, 37
In Datenbank aktivieren 14
Namespaces für 27
Parser 24, 37
Query-Strings 37, 47
Reihenfolge der Abfrageverarbeitung 39
URL-Textkodierung 26
XML-1.0-Spezifikation 22
XML-Daten aus Abfragen generieren 23
Zusammenfassung der Schritte für Zugriff auf XML-Daten 24
XML Custom Web Publishing 9
XML-Abfrage
Layout angeben 39
XML-Antwort
Layout wechseln 39
XML-Vorteile 9
<xsl:stylesheet>-Element 42
<xsl:template>-Element 42, 43

Z

Zugriffsprotokolldateien für Web-Server, Beschreibung 43
Zugriffsrechte 15
Zusammenfassung der Schritte für XML-Datenzugriff 24